

Владимир Паронджанов

# ЗАЧЕМ ВРАЧУ БЛОК-СХЕМЫ АЛГОРИТМОВ?

для визуализации медицинских знаний  
и предотвращения врачебных ошибок

*Новая система тщательно разработанных графических схем поможет вам научиться писать, читать и понимать медицинские алгоритмы при профилактике, диагностике, лечении, экстренной помощи, реанимации, реабилитации, прогнозе*

## Алгоритмы для эффективного клинического мышления

# **СОДЕРЖАНИЕ**

Введение. Алгоритмы — это совсем не больно!

## **Часть 1. КАК ОБЕСПЕЧИТЬ БЕЗОПАСНОСТЬ ПАЦИЕНТОВ? ВАЖНАЯ РОЛЬ МЕДИЦИНСКИХ АЛГОРИТМОВ**

Глава 1. Ошибки врачей. Могут ли алгоритмы спасти положение?

Глава 2. Почему медицина убивает и калечит пациентов?

Глава 3. Ахиллесова пята медицины и проблема неопределенности

Глава 4. Алгоритмы Гиппократа

## **Часть 2. ЗНАКОМЬТЕСЬ — МЕДИЦИНСКИЙ ЯЗЫК «ДРАКОН»**

Глава 5. Прыжок ДРАКОНа: из космоса в больничный коридор

Глава 6. Справочник: графические фигуры языка ДРАКОН

Глава 7. Простые медицинские алгоритмы. Правила и примеры

Глава 8. Логика в медицине и невидимая математика

Глава 9. Повторение медицинских действий, или цикл

Глава 10. Совместная работа врачей

Глава 11. Новый силуэт медицинского алгоритма

Глава 12. Сложные медицинские алгоритмы. Силуэт. Правила и примеры

Глава 13. Картографический принцип медицинского алгоритма (принцип красоты)

Глава 14. Алгоритм «Реанимация беременной женщины»

Глава 15. Медицинская алгоритмическая система

Глава 16. Какая польза от языка ДРАКОН для врачей?

Глава 17. ДРАКОН-конструктор. Как складывать алгоритм из кубиков?

## **Часть 3. ПЕРСПЕКТИВЫ АЛГОРИТМИЧЕСКОЙ МЕДИЦИНЫ**

Глава 18. Введение в алгоритмическую медицину. Теория медицинских алгоритмов.

Глава 19. Клиническое мышление должно опираться на медицинский алгоритмический язык

Глава 20. Русская медицинская инициатива

Глава 21. Медицинский язык на скамье подсудимых

Заключение

Литература

Список рисунков

Предметный указатель

Оглавление

## Введение

# АЛГОРИТМЫ — ЭТО СОВСЕМ НЕ БОЛЬНО!

## ЧТО МЫ ЗНАЕМ ОБ АЛГОРИТМАХ?

Мы живем в мире алгоритмов, хотя зачастую не догадываемся об этом. Современная цивилизация — это цивилизация алгоритмов. Они окружают нас повсюду.

Особенно важны медицинские алгоритмы, от качества которых зависит жизнь, здоровье и благополучие людей. К сожалению, многие врачи не умеют читать, писать и понимать алгоритмы. Впрочем, это дело поправимое.

Прочитав книгу, вы быстро получите нужные знания. Как известно, один рисунок стоит тысячи слов. К вашим услугам — четкие, кристально ясные и эргономичные рисунки, сделанные так, чтобы читатель «Посмотрел — и сразу понял!». Они помогут открыть заветную дверь в увлекательное царство медицинских алгоритмов.

## В ЧЕМ ПРОБЛЕМА?

Трудность в том, что алгоритмы диагностики и лечения, как правило, очень сложны. Понять их непросто. Нужно изрядно попотеть, затратить много труда и времени.

А нельзя ли найти обходную дорогу и сэкономить время?

Конечно, можно! Секрет в том, что алгоритмы надо сделать *дружелюбными* (people-friendly). Это позволит превратить головоломки в наглядные алгоритмы-картинки, обеспечивающие быстрое и глубокое понимание.

Почему алгоритмы трудны для понимания? Потому что существующие способы записи медицинских алгоритмов устарели и превратились в досадное препятствие. Они созданы без оглядки на требования науки о человеческих факторах — эргономики. Эти устаревшие способы не учитывают психофизиологические характеристики человека. И тем самым затрудняют и замедляют работу с алгоритмами.

## ЛЕГКИЕ ДЛЯ ПОНИМАНИЯ И УДОБНЫЕ ДЛЯ РАБОТЫ

В книге излагается новый взгляд на медицинские алгоритмы. Мы покажем, что алгоритмы должны быть не только правильными, но и *дружелюбными*. То есть легкими для восприятия, удобными для запоминания и полезными для анализа.

Этой благородной цели служит эргономичный алгоритмический язык.

Многие врачи вздрагивают при словах «алгоритмический язык». Им кажется, что это что-то ужасное и далекое от медицины.

На самом деле, это не так. Давайте послушаем Петю и Лену.

*Врач Петя.* Как следует записывать медицинские алгоритмы? По правилам? Или наобум, без всяких правил?

*Доктор Лена.* Конечно, по правилам.

*Врач Петя.* Я хочу рисовать алгоритмы строго по правилам. Где их взять?

*Доктор Лена.* Правила записаны в языке. Язык задает правила.

*Врач Петя.* Какой язык?

*Доктор Лена.* Язык для записи алгоритмов.

*Врач Петя.* Как же называется этот язык?

*Доктор Лена.* Он называется «медицинский алгоритмический язык».

Прочитав книгу, ты узнаешь все правила.

Хорошие алгоритмы создают повышенный интеллектуальный комфорт для врачей, увеличивают эффективность лечебно-диагностического процесса.

Алгоритмы являются неотъемлемой частью человеческой и медицинской культуры. Умение составлять и анализировать алгоритмы позволяет улучшить работу ума врачей. И, что очень важно, избежать многих серьезных ошибок.

Как известно, медицинские ошибки являются болевой точкой медицины, их предотвращению уделяется большое внимание.

## БЕЗОПАСНОСТЬ ПАЦИЕНТОВ

Журден, герой Мольера, не знал, что всю жизнь говорил прозой. Многие врачи не знают, что всю жизнь выполняют медицинские алгоритмы.

А иногда, увы, не выполняют. Тогда происходят врачебные ошибки.

Раньше считалось, что ошибки — внутреннее дело медицины, поучительный материал, чтобы публично или келейно пристыдить виновных. Но теперь ситуация изменилась.

Медицинский мир с удивлением обнаружил, что дефекты медицинской помощи касаются не только врачей, но и больных. Выяснилось, что упущения врачей могут повлечь за собой смерть, стойкую инвалидность и иные неприятности для пациентов.

Конечно, все это было известно и раньше, однако медики не видели здесь проблемы. Происходили трагедии, пациенты погибали, однако даже лучшие врачи мира не могли найти выход из тупика. Лишняя ошибка и лишняя смерть трактовались как печальные, но неизбежные издержки профессии. Проблема существовала, но была скрытой, неявной, неосознанной.

Первый раскат грома грянул 1999 году, когда были обнаружены вопиющие факты и обнародована статистика потерь. В трех докладах Института медицины Национальной академии наук, техники и медицины США, опубликованных в 2000, 2001 и 2015 годах, со всей остротой поставлен вопрос о безопасности пациентов. И предложен ряд мер для их защиты.

Ценность докладов Института медицины состоит в том, что постановка проблемы отличается принципиальной новизной. В двух словах ее можно выразить так:

В чем проблема?

Проблема в том, что медицина убивает и калечит пациентов

Смертность по вине врачей в больницах США очень велика и является одной из ведущих причин в структуре смертности населения. Пациенты больше не согласны отправляться в морг из-за того, что врачи допускают предотвратимые (preventable) ошибки [1] [2] [3].

Дело получило огласку и вырвалось за пределы врачебных кабинетов. По рекомендации Института медицины, в конгрессе США были проведены слушания и принят закон о безопасности пациентов (Patient Safety and Quality Improvement Act of 2005), подписанный президентом Джорджем Бушем младшим 29 июля 2005 года [4, 5].

## ПРОБЕЛ В ДОКЛАДАХ ИНСТИТУТА МЕДИЦИНЫ

Выводы и предложения Института медицины являются важными, но недостаточными.

Врачебные ошибки зависят от многих причин, в том числе, от недостатков медицинского языка, который, будучи естественным языком, не приспособлен для точного и удобного описания медицинских алгоритмов и не имеет необходимых для этого специальных средств.

Алгоритмы профилактики, диагностики, лечения, скорой помощи, реанимации, реабилитации, прогноза являются научной проблемой первостепенной важности, которая имеет прямое отношение к предотвращению врачебных ошибок и безопасности пациентов. Однако эта проблема была полностью упущена из виду в докладах Института медицины, что снижает ценность его выводов и рекомендаций.

## МНОГОГОЛОВАЯ ГИДРА

Чем же кончилось дело? Решен ли вопрос о безопасности больных?

Безопасность пациентов — исключительно сложная, многофакторная проблема, многоголовая гидра. Решить ее непросто. На быстрое решение рассчитывать не приходится. Это будет длительный и трудный процесс.

Одна из возможных причин — низкое качество медицинских алгоритмов, которые создаются без детальной проработки всех возможных маршрутов и разветвлений, что и порождает врачебные ошибки. Другая причина — неполнота или отсутствие необходимых алгоритмов.

Похоже, что проблема медицинских алгоритмов выпала из поля зрения ученых, как медиков, так и математиков. Она не рассматривается как фундаментальная проблема медицины, считается малозначимой и не изучена в должной мере. Многие врачи по-прежнему не догадываются, что живут в мире медицинских алгоритмов.

Дело усугубляется тем, что некоторые медики относятся к алгоритмам с недоверием и предубеждением. На их знамени написано: «Медицина относится к разряду наук неточных, железных алгоритмов в ней нет и быть не может» [6].

## АЛГОРИТМЫ — НОВЫЙ СПОСОБ ИЗЛОЖЕНИЯ МЕДИЦИНСКИХ ЗНАНИЙ

— Ну и что? Врачи иногда допускают ошибки, вследствие этого пациенты иногда умирают, все это верно. Но, черт возьми, причем здесь алгоритмы? — спросит рассерженный читатель.

Книга, которую вы держите в руках, является обстоятельным и аргументированным ответом.

Алгоритмы — новый, значительно более удобный и эффективный способ представления медицинских знаний<sup>1</sup>. Их можно и нужно рассматривать как новую парадигму медицинского образования.

— В чем же его эффективность?

— В частности, в том, что алгоритмы защищают врача от претензий со стороны пациентов.

— Это каким же образом?

— Алгоритмы вносят ясность в запутанные подробности лечения. А там, где есть ясность, вероятность ошибки мала. Чем меньше ошибок, тем счастливее врач.

— Неправда! В медицинских книгах много разговоров об алгоритмах, а воз и ныне там. Никакой ясности от этого не прибавилось. Почему?

— Потому что в книгах есть отдельные недостатки. Медицинские алгоритмы описаны шиворот-навыворот, через пень-колоду, с грубыми алгоритмическими и эргономическими ошибками.

— Причем тут алгоритмические ошибки? Вы что, замыслили повесить на врачей еще и программирование?

— Ни в коем случае. Программирование вообще ни при чем. Речь идет лишь о том, что медицинские алгоритмы должны быть легкими для понимания и иметь красивую упаковку. Чтобы любой врач и студент-медик мог быстро понять и запомнить алгоритм. В нынешних условиях это невозможно. Сегодня алгоритмы пишут по принципу «умрешь не поймешь».

— Уж не хотите ли вы сказать, что в вашей книге предложен лучший в мире способ описания медицинских алгоритмов? И что именно этот способ следует сделать международным медицинским стандартом?

Автор скромно потупил глаза.

## ОСНОВНАЯ МЫСЛЬ

Профессиональный медицинский язык (язык медицинской литературы, учебников, стандартов, руководств, клинических рекомендаций, протоколов) имеет серьезный недостаток. Он недостаточно точен и плохо приспособлен для описания сложных и разветвленных, нередко многочасовых и многодневных медицинских действий, решений и процедур, выполняемых при профилактике, диагностике, лечении, экстренной помощи, реанимации, реабилитации, прогнозе.

Чтобы устранить дефект, нужно осуществить глубокую реформу медицинского языка, расширив его возможности с помощью визуального медицинского алгоритмического языка. Последний предназначен для стимулирования клинического мышления врачей, повышения безопасности пациентов, предотвращения врачебных ошибок и стандартизации представления медицинских алгоритмов в медицинской литературе.

## СВЕРХЗАДАЧА

Ради чего написана книга? Какой социально значимый и общественно полезный результат мы хотим получить? Какова сверхзадача и стратегическая цель?

Книга преследует несколько целей. Среди них в качестве наиболее важных можно выделить две: *безопасность пациентов и эффективность образования*:

<sup>1</sup> Имеются в виду не декларативные, а процедурные знания.

Стратегическая  
цель книги

- обеспечить более высокий уровень безопасности пациентов,
- повысить эффективность медицинского образования.

## ПРИНЦИП «НЕВИДИМОЙ» МАТЕМАТИКИ

Если использовать в алгоритмах математические формулы, проект сразу же потерпит крах. Врачи не любят формулы и не станут их читать — они проголосуют ногами. Поскольку книга предназначена для врачей, выход один — придется обойтись без формул.

Можно ли создать математику без формул? Раньше это считалось почти невозможным. Однако голь на выдумки хитра. Пролистайте страницы. Вы не встретите ни одной формулы.

Тем не менее, все медицинские алгоритмы, представленные в книге, опираются на строгий математический фундамент. Фокус в том, что математика спрятана в графике, она стала невидимой.

Принцип «невидимой» математики означает, что врачи могут читать книгу со спокойной душой, не испытывая ни малейших затруднений.

## ЛЕЙТМОТИВ КНИГИ

В качестве лейтмотива выбрана цитата:

«Несмотря на растущее техническое оснащение медицинских учреждений, повышение квалификации врачей и успехи медицинской науки, количество больных, пострадавших от дефектов медицинской помощи, во всех странах нарастает» [178]

Это очень важная мысль. Вместе с тем она не бесспорна; возможно, кто-нибудь захочет подвергнуть ее сомнению. Действительно, во многих государствах до сих пор не ведется статистика подобных дефектов. Поэтому утверждать, что «во всех странах нарастает» число жертв врачебных ошибок, строго говоря, неправомерно.

Однако подобные неточности не должны заслонять главное. Нет сомнения, что проблема ошибок приобрела всемирный характер и бросает зловещую тень на всю медицину. А раз так, значит, больше нельзя закрывать глаза на факты и прятать голову в песок. Проблема нуждается в неотложном решении.

## АЛГОРИТМИЧЕСКАЯ МЕДИЦИНА

Решение должно быть глубоким и всесторонним. Полумерами вряд ли можно обойтись. Нынешняя медицина, по-видимому, должна научиться говорить на новом языке — языке алгоритмов. Медицина будущего — это алгоритмическая медицина.

Медицина должна изменить свой облик и превратиться в алгоритмическую медицину.

Программу обучения в медицинских университетах следует усовершенствовать на основе принципов алгоритмической медицины.

## **«УЛИЦА КОРЧИТСЯ БЕЗЪЯЗЫКАЯ — ЕЙ НЕЧЕМ КРИЧАТЬ И РАЗГОВАРИВАТЬ!»**

Если у человека нет языка, он не сможет говорить и становится немым.

Сегодня медицина — это Великий Немой, который хочет, но не может шагнуть в век алгоритмизации. Потому что у нее нет языка. По меткому выражению Маяковского, «ей нечем кричать и разговаривать».

Многие врачи хотят описать свои знания в виде алгоритмов, но, увы, получается плохо и коряво. В итоге возникает утешительный самообман, появляется привычка называть алгоритмом то, что алгоритмом вовсе не является.

Почему? Потому что нет языка для записи алгоритмов.

Отсутствие языка, отсутствие удобной нотации для записи медицинских алгоритмов, нотации, которая отвечает чаяниям врачей и облегчает их труд, является главным, основополагающим препятствием, которое мешает движению вперед

## **ПОМОЩЬ УЖЕ В ПУТИ. МЕДИЦИНСКИЙ ДРАКОН**

К счастью, в России создан графический язык ДРАКОН, позволяющий представлять алгоритмы в виде наглядных картинок. Он нравится врачам и рекомендуется для широкого применения.

Почему ДРАКОН? Ответ простой: змея — символ медицины, а дракон — это змея с крыльшками.

Если у вас возникли трудности и вы совсем забыли, что такое алгоритм, не беда — скачайте и посмотрите детскую книжку «Волшебный Дракон в гостях у Мурзика» [7]. И неясности сразу исчезнут<sup>2</sup>.

Или задайте вопрос на форуме ДРАКОНа [8].



## **ИГРА В ДЕТСКИЕ КУБИКИ И КОНСТРУКТОР АЛГОРИТМОВ**

Дети любят играть в кубики. Язык ДРАКОН позволяет врачам совместить приятное с полезным. Медицинские алгоритмы можно создавать играющи, по примеру детской игры. Для этого существует медицинский ДРАКОН-конструктор.

<sup>2</sup> «Занимательная информатика, или Волшебный Дракон в гостях у Мурзика». В аннотации сказано: «Эта умная и веселая книга посвящена алгоритмам и принципам их построения. Просто и, что самое главное, очень доступно для детей, автор знакомит их с волшебным миром алгоритмов. С первых страниц ребенок попадает в мир игры и сказочных героев, которые помогают ему... Книгу можно рекомендовать родителям для совместного чтения с детьми и школьникам для развития интеллекта» [184].

Скачать [http://drakon.su/\\_media/biblioteka\\_1/zanim\\_inf\\_optimizacija1\\_.pdf](http://drakon.su/_media/biblioteka_1/zanim_inf_optimizacija1_.pdf) .

Конструктор алгоритмов — надежный помощник врача. Он умело подсказывает, как нужно составлять алгоритмы. Словно посох мудреца, он контролирует каждый ваш шаг, не дает оступиться и сбиться с пути.

Желающие могут сразу опробовать услугу и пройти онлайн-тренинг: <https://drakon-editor.com/> Однако мы рекомендуем не торопиться и сначала все-таки ознакомиться с текстом. Подробнее см. стр. 127.

## **В ДОБРЫЙ ПУТЬ С ДРАКОНОМ В РЮКЗАКЕ!**

Прочитав книгу, читатель сможет легко ориентироваться в империи медицинских алгоритмов. И убедиться, что медицинский язык ДРАКОН — удобное средство, помогающее профессиональным врачам и фармацевтам писать, читать и понимать алгоритмы.

Используя интуитивно понятный ДРАКОН-конструктор, инициативные медработники на местах сумеют самостоятельно, без чьей-либо помощи, создавать наглядные и понятные медицинские алгоритмы. Вы сможете делиться знаниями друг с другом, советоваться с коллегами и получать рецензии. Ваши предложения в виде ДРАКОН-алгоритмов можно вынести на всеобщее обсуждение — на научных конференциях, в рабочих группах (workshop) и через Интернет.

В результате медицина приобретает новый мощный канал для самоуправления и самосовершенствования.

## **МЕДИЦИНСКИЕ БИЗНЕС-ПРОЦЕССЫ**

Организаторы и руководители здравоохранения получают удобный инструмент для управления медициной в виде бизнес-процессов на языке ДРАКОН. Это принесет пользу на всех уровнях медицинской иерархии. И на уровне лечебно-профилактического учреждения, и на уровне регионов, и на уровне всей страны.

Каким образом министр здравоохранения может контролировать свое огромное хозяйство, начиная от сельского здравпункта в таежной глубинке и кончая аппаратом Министерства?

Образно говоря, нужна сеть «всевидящих видеокамер». Беда в том, что бизнес-процессы абстрактны и недосягаемы для камер. Этому горю нетрудно помочь. Дракон-алгоритмы играют роль волшебных видеокамер. Они срывают шапку-невидимку с бизнес-процессов и делают их доступными для государева ока.

Под пристальным взглядом ДРАКОНа все тайное становится явным. Все бизнес-процессы предстают перед управлением и контролирующими органами полностью обнаженными — во всей своей бесстыдной наготе.

Благодаря ДРАКОНу любая оплошность видна как на ладони. Руководитель сразу видит, где слабое место, и быстро устраняет управленческий прокол.

## **СТРУКТУРА КНИГИ**

Книга состоит из трех частей. В первой части (главы 1–4) обсуждается возможность использования графического алгоритмического языка ДРАКОН для решения проблемы безопасности пациентов и повышения эффективности медицинского образования.

Вторая часть (главы 5–10) содержит подробное описание графического языка ДРАКОН, который рекомендуется для широкого применения в медицине. Приводятся многочисленные примеры медицинских дракон-алгоритмов.

Третья и заключительная часть (главы 11–14) посвящена обсуждению новых результатов и перспектив, которые открываются перед медициной и медицинским образованием благодаря стандартизации медицинских алгоритмов.

Часть 1

# **КАК ОБЕСПЕЧИТЬ БЕЗОПАСНОСТЬ ПАЦИЕНТОВ? ВАЖНАЯ РОЛЬ МЕДИЦИНСКИХ АЛГОРИТМОВ**

## Глава 1

# ОШИБКИ ВРАЧЕЙ. МОГУТ ЛИ АЛГОРИТМЫ СПАСТИ ПОЛОЖЕНИЕ?

### ЗАЧЕМ НУЖНА ЭТА КНИГА?

Цель книги — обосновать необходимость реформы медицинского языка и предложить средство для ее практического осуществления. Таким средством является алгоритмизация медицинской литературы, широкое использование медицинских алгоритмов *высокой точности* в медицинских учебниках, стандартах, руководствах, клинических рекомендациях, протоколах.

Неприятность в том, что сегодня в медицинских изданиях почти повсеместно применяются неточные, приблизительные, неудовлетворительные описания медицинских алгоритмов, выполняемые с помощью естественного языка и неквалифицированных рисунков. Они написаны с явным нарушением алгоритмических и эргономических правил, что вносит путаницу, затрудняет восприятие, осмысление и понимание алгоритмов.

Подобная ситуация недопустима, так как любая неточность в учебниках и иных публикациях может привести к врачебным ошибкам, что отрицательно сказывается на здоровье населения.

Чтобы поправить дело, необходимо:

- решительно отказаться от существующей практики, с неизбежностью порождающей ошибки,
- осуществить глубокую реформу медицинского языка, расширив его возможности с помощью специального визуального медицинского алгоритмического языка *высокой точности*.

Такой язык реально существует, он разработан в Федеральном космическом агентстве России, прошел апробацию, описан в данной книге и выносится на суд читателей.



Рис. 1. Медицинский алгоритм «Действия при травматической ампутации частей тела» [179].

## **ВРАЧЕБНЫЙ ЯЗЫК ПРИНОСИТ БЕДУ**

Современный медицинский язык имеет принципиальный недостаток. Как и любой естественный язык, он не обеспечивает необходимую точность описания сложных вопросов. По этой причине он не гарантирует защиту от врачебных ошибок, а наоборот, провоцирует их появление. В связи с этим, медицинский язык представляет серьезную опасность для пациентов. Он порождает ошибочные действия медперсонала, которые могут привести и зачастую приводят к смерти, инвалидности или иному ущербу для больных.

Как известно, современная медицина рассматривает повышение безопасности пациентов как приоритетную задачу [1] [2] [3]. Выявлены и частично устраниены многие причины этого негативного явления. Однако проблема оказалась сложнее, чем предполагалось, и известные решения не полностью закрывают вопрос.

К сожалению, проблема «медицинский язык как источник неприятностей» почти полностью выпала из поля зрения ученых. Она не рассматривается как фундаментальная проблема медицины, считается несущественной, недооценивается и не изучена в должной мере.

В книге впервые проводится тщательное исследование влияния языка на врачебные ошибки и безопасность пациентов, выявляются прочие серьезные недостатки медицинского языка. Выдвигаются конкретные, тщательно продуманные и вместе с тем простые практические предложения, пригодные для массового применения.

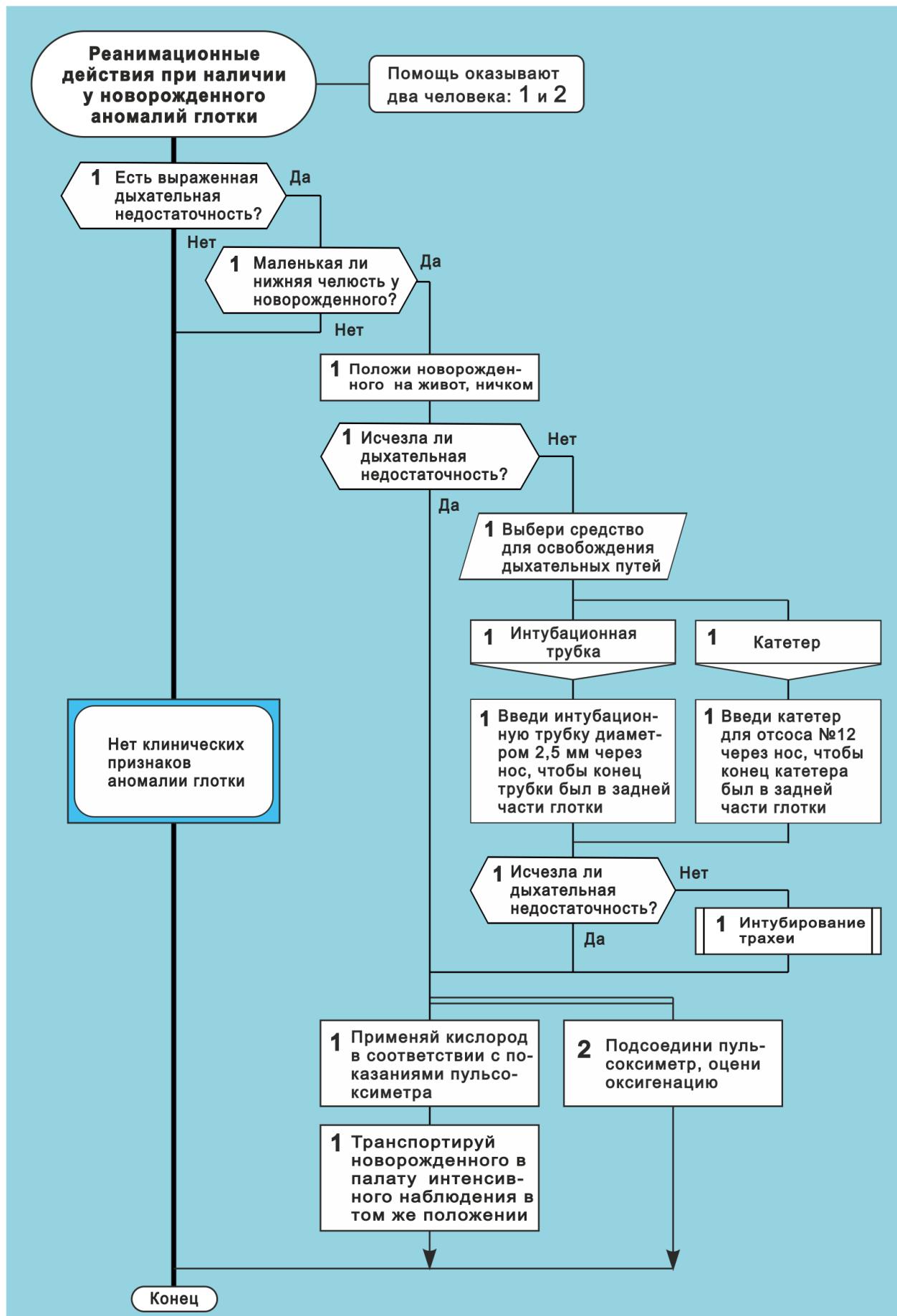
## **ЧТО ПРЕДЛАГАЕТСЯ?**

Предлагаются средства и метод, позволяющие решить три крупных задачи:

1. Реформировать профессиональный медицинский язык, (язык медицинской литературы, учебников, стандартов, руководств, клинических рекомендаций, протоколов), дополнив его медицинским алгоритмическим языком высокой точности.
2. Улучшить форму представления медицинских знаний в медицинской литературе, устранить неточность, приблизительность и неполноту описаний — все это можно и нужно сделать с помощью медицинских алгоритмов высокой точности.
3. Изменить методику преподавания медицинских дисциплин в медицинских университетах и системе последипломного образования, используя алгоритмы высокой точности.

Центральным пунктом реформы является развитие клинического мышления врачей, которое превращается в *алгоритмическое клиническое мышление высокой точности*.

Выигрыш достигается за счет визуализации, формализации и стандартизации пошагового представления медицинских алгоритмов на бумаге и на экране компьютера.



**Рис. 2.** Медицинский алгоритм «Реанимационные действия при наличии у новорожденного аномалий глотки» [9].

## ФАЛЬШИВО И НЕУДАЧНО

Философ Уиллард Куайн как-то сказал, что язык можно представить в образе корабля, находящегося в море и нуждающегося в ремонте [10]. Эту метафору можно применить и к медицинскому языку, который словно корабль странствует в медицинских морях. И тоже нуждается в починке.

Изрядная «пробоина» в медицинском корабле образовалась, когда врачи стали необоснованно использовать слово алгоритм, плохо понимая его смысл. Получилось фальшиво и неудачно. Чтобы поправить дело, надо устраниТЬ фальшЬ и внести гармонию в язык врачей с помощью визуального медицинского алгоритмического языка.

## ЧТО ТАКОЕ МЕДИЦИНСКИЕ АЛГОРИТМЫ?

Грубо говоря, это все то, что делают врачи. Например, при пересадке почки, лечении туберкулеза или экстренном оказании помощи жертвам мощного взрыва, землетрясения или иного бедствия.

Медицинские алгоритмы — это любые врачебные процедуры, решения, действия, а также составленные из них замысловатые и разветвленные цепочки операций, выполняемые при профилактике, диагностике, лечении, скорой помощи, реанимации, реабилитации, прогнозе<sup>3</sup>.

Уточним: алгоритмами являются не сами действия и решения, а их точные описания на бумаге или экране.

Медицинские алгоритмы есть точное описание всех без исключения медицинских действий и решений, а также пошаговое описание их последовательностей

Можно также сказать, что алгоритмы — простой, удобный и эффективный язык для общения врачей и для обмена медицинскими знаниями.

## А ЧЕМ МЫ ХУЖЕ?

Слово «алгоритмы» пришло в медицину недавно. Конечно, врачи и раньше спасали и выхаживали больных, однако их повседневная работа была безымянной, не имела звучного названия. Но свято место пусто не бывает. Нашелся безвестный медик-энтузиаст, который воскликнул: а чем мы хуже? У всех порядочных людей есть алгоритмы, а в медицине их почему-то нет.

Сказано — сделано. Еще вчера во всем мире не было медицинских алгоритмов, а сегодня, пожалуйста, они выросли повсюду, как грибы после дождя. И мигом заполнили медицинские учебники, руководства, клинические рекомендации, протоколы.

При этом, по сути почти ничего не изменилось. Появилось лишь модное словечко «алгоритм», щедро рассыпанное в нужных местах. И прежде безликий медицинский текст сразу как-то преобразился, подбоченился и заиграл новыми красками. Вот пример: *алгоритм ведения пациента при нормотензивной глаукоме* [11].

<sup>3</sup> В широком смысле к медицинским алгоритмам относятся также действия, решения и составленные из них цепочки операций, выполняемые при фармации, фармакологии, ветеринарии, а также при организации и управлении здравоохранением.

## ВОЛЬНИЦА КОНЧИЛАСЬ

Казалось бы, одно слово мало что значит. Однако это не так. Алгоритм — символ образцового порядка, строжайшей дисциплины и жесткой стандартизации. В течение долгих тысячелетий врачи лечили по заветам предков, проявляли творчество, полагались на свой опыт и обходились без стандартов.

А тут вдруг — бац! — вольница кончилась, надо уважать стандарты и соблюдать алгоритмы. Многие к этому не готовы. К тому же качество алгоритмов оставляет желать лучшего.

## ЧТО ЛУЧШЕ: ТЕКСТ ИЛИ ГРАФИКА?

Жизнь не стоит на месте. Некоторые медики полагают, что алгоритмы, описанные текстом, это вчерашний день медицины. По их мнению, алгоритмы следует изображать в виде наглядных графических рисунков. Имеются в виду блок-схемы алгоритмов, диаграммы деятельности языка UML, деревья принятия решений [12, 13] [14] [15] [16] [17] [18] [19] [20] [21] [22] [23] [24] [25].

Вот мы и подошли к центральной, узловой проблеме, над которой нынче бьются пытливые умы. Как лучше описывать алгоритмы? С помощью текста? Или с помощью графики?

## КОНФЛИКТ МИРОВОЗЗРЕНИЙ

Основной закон алгоритмизации в вольном пересказе выглядит примерно так: *алгоритм следует описывать мелкими шагами, т. е. шаг за шагом, очень подробно и ничего не пропуская*. Увы, медицинские писатели сплошь и рядом нарушают этот закон. Почему? Может быть, писатели плохие? Вовсе нет. Писатели хорошие, знающие, есть среди них и опытные врачи, и преподаватели медвузов, и ученые. Так в чем же дело?

Здесь мы с удивлением обнаруживаем очень интересное, в чем-то парадоксальное явление, которое можно охарактеризовать как своеобразный *конфликт мировоззрений или конфликт двух культур, как столкновение математической культуры мышления с культурой медицинской*.

Они сошлись. Волна и камень,  
Стихи и проза, лед и пламень  
Не столь различны меж собой...

## САМИ С УСАМИ

В математике господствует формальный язык, а медицина считается плохо формализуемой областью знания. В этом глубинная суть конфликта. Математические алгоритмы любят строгость, причем такую, которую нельзя выразить с помощью естественного языка. Чтобы записать строгий алгоритм, нужен не естественный, а формальный, искусственный язык, например, алгоритмический язык программирования.

Как же быть? Не желая вникать в математические тонкости, потому как сами с усами, медики, недолго думая, попытались скрестить ужа с ежом и, к сожалению, соединили несоединимое. Почему так получилось?

## КОСТЬ В ГОРЛЕ

В медицине, как и в других науках, существуют некие незыблемые устои, образовавшиеся в ходе длительного исторического развития. Это стереотипы медицинского мышления, устоявшиеся профессиональные привычки, наработанный стиль написания медицинской литературы.

Понятие алгоритма вступает в острое противоречие с идеалами и нормами медицинского мышления; оно торчит в тексте, как кость в горле. Игнорируя это обстоятельство и действуя методом «грубой силы», медики механически внедрили слово «алгоритм» в традиционную структуру медицинских текстов. Фактически они осуществили своеобразный «интеллектуальный киднэпинг» — похитив у математиков понятие алгоритма, они искалечили его почти до неузнаваемости. И незаконно присвоили себе право распоряжаться чужой собственностью (краденым) по своему усмотрению.

## ЧУДО В ПЕРЬЯХ

Что же получилось в результате сенсационной кражи века? Получился некий загадочный кентавр, чудо в перьях, который имеет лишь запах алгоритма.

«Что же из этого следует? — Следует жить» и, по возможности, навести порядок в авгиевых конюшнях медицинского языка. Укажем два важных пункта.

1. Сегодня подавляющее большинство медицинских алгоритмов описано в литературе в виде текста на естественном языке. Это недопустимо, потому что обычный язык *категорически не пригоден* для записи безошибочных алгоритмов. Здесь находится скрытый источник врачебных ошибок.

2. Ничтожно малая часть (крохотные доли процента) описана в виде графических рисунков. Но и здесь дело не ладится — нынешняя графика подходит лишь для самых простых задач.

## ГОРА РОДИЛА МЫШЬ

С ростом сложности графические схемы алгоритмов быстро теряют наглядность, линии начинают пересекаться и сплетаются в невразумительный клубок. Чтобы сохранить удобочитаемость, приходится упрощать и сокращать чертеж. Полноценный алгоритм превращается в усеченный, неточный, куцый вариант.

Задача пошагового описания медицинских алгоритмов в удобной для врачей форме как была, так и осталась нерешенной.

## РОССИЯ НАМ ПОМОЖЕТ

Вместе с тем уже созрели предпосылки для успешного решения проблемы. В России создан графический алгоритмический язык, который выгодно отличается от предшественников. Он позволяет:

- представить сложные и сверхсложные медицинские алгоритмы не в сокращенной форме, а полностью, без пропусков и упрощений;
- использовать эргономичную, удобную для врачей форму представления алгоритмов;
- устранить основной недостаток профессионального медицинского языка, т. е. обеспечить подробное (пошаговое) описание сложных и разветвленных медицинских алгоритмов в точной, однозначной и исчерпывающей форме.

## МЕЖДУ СЦИЛЛОЙ И ХАРИБДОЙ: ЗАЧЕМ ВРАЧУ АЛГОРИТМИЧЕСКИЙ ЯЗЫК?

На каком языке следует писать медицинские алгоритмы? Язык — это система правил. Если правила хорошие, язык обречен на успех. Если плохие, языком никто не станет пользоваться. И он благополучно умрет.

Какие требования надо предъявить к языку для записи алгоритмов?

*Во-первых*, он должен быть формально строгим. Это нужно, чтобы свести к минимуму врачебные ошибки. Обычный медицинский язык не годится для этой цели: он прячет и маскирует ошибки, вместо того, чтобы предотвращать и исключать их.

*Во-вторых*, нужна удобочитаемость, легкость понимания. Зачем? Чтобы создать для медиков удобные и комфортные условия. Врач не желает ломать зубы о несъедобный гранит науки. Он хочет легко и быстро схватить суть алгоритма — с первого взгляда. По принципу: «Посмотрел и сразу понял!» Все алгоритмы в этой книге нарисованы именно так.

Второе требование намного важнее первого. Потому что профессиональный врач — это высший судья. Если врачи скажут: «язык трудный и неудобный», значит, проект провалился.

Чтобы избежать такого финала, в книге предложен очень легкий медицинский алгоритмический язык. Взглядите на рисунки 1 и 2. Все надписи даны на знакомом для медиков языке. Вся терминология медицинская. Нет ни одного чужого слова. Чтобы читать алгоритм, ничего кроме медицины, знать не нужно.

А где же формальная строгость, где математика? Она убрана с глаз долой и спрятана в графике. Это сделано для того, чтобы не отвлекать внимание врача на ненужные ему детали.

Но это вовсе не значит, что математика исчезла. Математика есть, она играет очень важную роль, так как графический каркас алгоритма строится по математическим законам (подробнее см. главы 4 и 12).

## МЕДИЦИНСКИЕ АЛГОРИТМЫ ВЫСОКОЙ ТОЧНОСТИ

Итак, мы предлагаем принципиально новый тип медицинских алгоритмов.

Как отличить новый тип от того, что было прежде? Нужен отличительный признак. Для этого вводится термин «медицинский алгоритм высокой точности» или, что одно и то же, медицинский алгоритм на языке ДРАКОН.

Слова *высокая точность* подчеркивают строгость алгоритма и тот факт, что он обоснован математически [26]. Они напоминают, что алгоритмы нового типа позволяют с ювелирной точностью изобразить любой, сколь угодно сложный медицинский алгоритм.

Вместе с тем, необходимо обеспечить удобства для врача, который привык к естественному языку и не хочет от него отказываться. Для этого используется хитрый прием. *Математически строгая графика элегантно сочетается с надписями на обычном языке* (см. рис. 1 и 2).

Подобное сочетание оказывается плодотворным, так как медицинские алгоритмы высокой точности приобретают два полезных свойства. К красоте математики добавляется прелест эргономики, то есть знакомые сообщения на родном языке.

- Медицинский язык ДРАКОН специально создан для построения медицинских алгоритмов высокой точности.
- Язык ДРАКОН называется медицинским алгоритмическим языком высокой точности.

## ЧТО ТАКОЕ ШАГ АЛГОРИТМА?

Алгоритм делится на мелкие порции — шаги, которые следуют друг за другом (шаг за шагом).

На рис. 3 показан алгоритм, состоящий из пяти шагов. На каждом шаге выполняется одно медицинское действие. Действия выполняются последовательно, друг за другом и решают поставленную задачу. Заголовок и Конец (закругленные фигуры) — это обрамление алгоритма; они не выполняют никаких действий.

Можно сказать по-другому: алгоритм есть графическая инструкция для врача. На рис. 3 представлена инструкция, содержащая пять команд (шагов), которые должен выполнить врач.

## ЧТО ТАКОЕ МЕДИЦИНСКОЕ РЕШЕНИЕ?

На рис. 3 рассмотрен простейший случай — все шаги описывают действия. На рис. 4 изображена более сложная ситуация, при которой врач использует не только действия, но и медицинские решения.

Мы вынуждены принимать решение всякий раз, когда на дороге встречается развилка. Куда идти? Направо или налево? «Налево пойдешь — коня потеряешь, направо пойдешь — голову сложишь».

Нумерация шагов зависит от принятых решений. Предположим, что, следуя по дороге, мы всегда (на каждом перекрестке) будем сворачивать направо. Применяя принцип «на развилке всегда направо» к рис. 4, можно убедиться, что «правая дорога» состоит из 10 шагов.

Первый шаг — решение «Есть ли выраженная дыхательная недостаточность?». Решение состоит в том, что мы выбираем ответ «Да».

Второй шаг — «Маленькая ли челюсть у новорожденного?». Врач смотрит на малютку и опять решает «Да».

Третий шаг — действие: «Положи новорожденного на живот, ничком».

Четвертый шаг снова решение: «Исчезла ли дыхательная недостаточность?» Врач смотрит и видит: нет, не исчезла.

Пятый шаг — еще раз нужно принять решение. Какой инструмент будем использовать: интубационную трубку или катетер? Обратите внимание: решение в данном случае изображается с помощью трех фигур в белой рамке.

Пропустим несколько шагов и взглянем на шаг 9. Он включает два действия, которые выполняются одновременно. Ассистент подсоединяет пульсоксиметр и проверяет оксигенацию, а врач подает кислород, контролируя показания прибора.

Таким образом, мы выяснили: «правая дорога» на рис. 4 содержит десять шагов. Из них пять — медицинские действия и пять — медицинские решения.

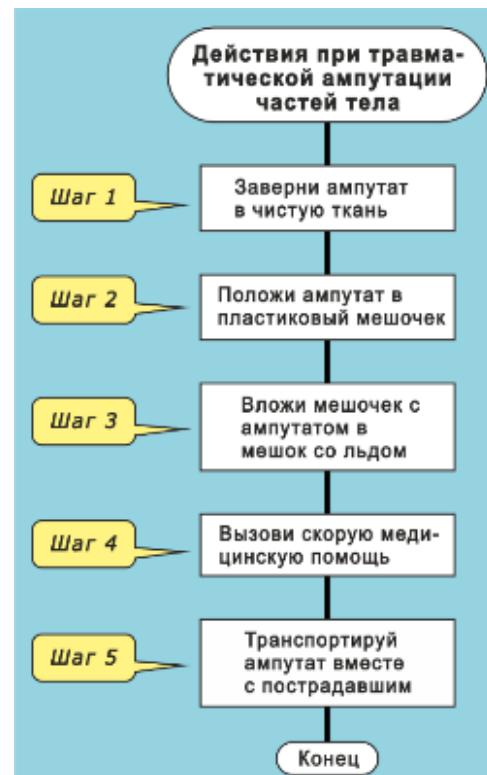


Рис. 3. Алгоритм состоит из пяти шагов

Что такое шаг алгоритма

- Это действие или решение.
- Один шаг — одно медицинское действие или одно медицинское решение.

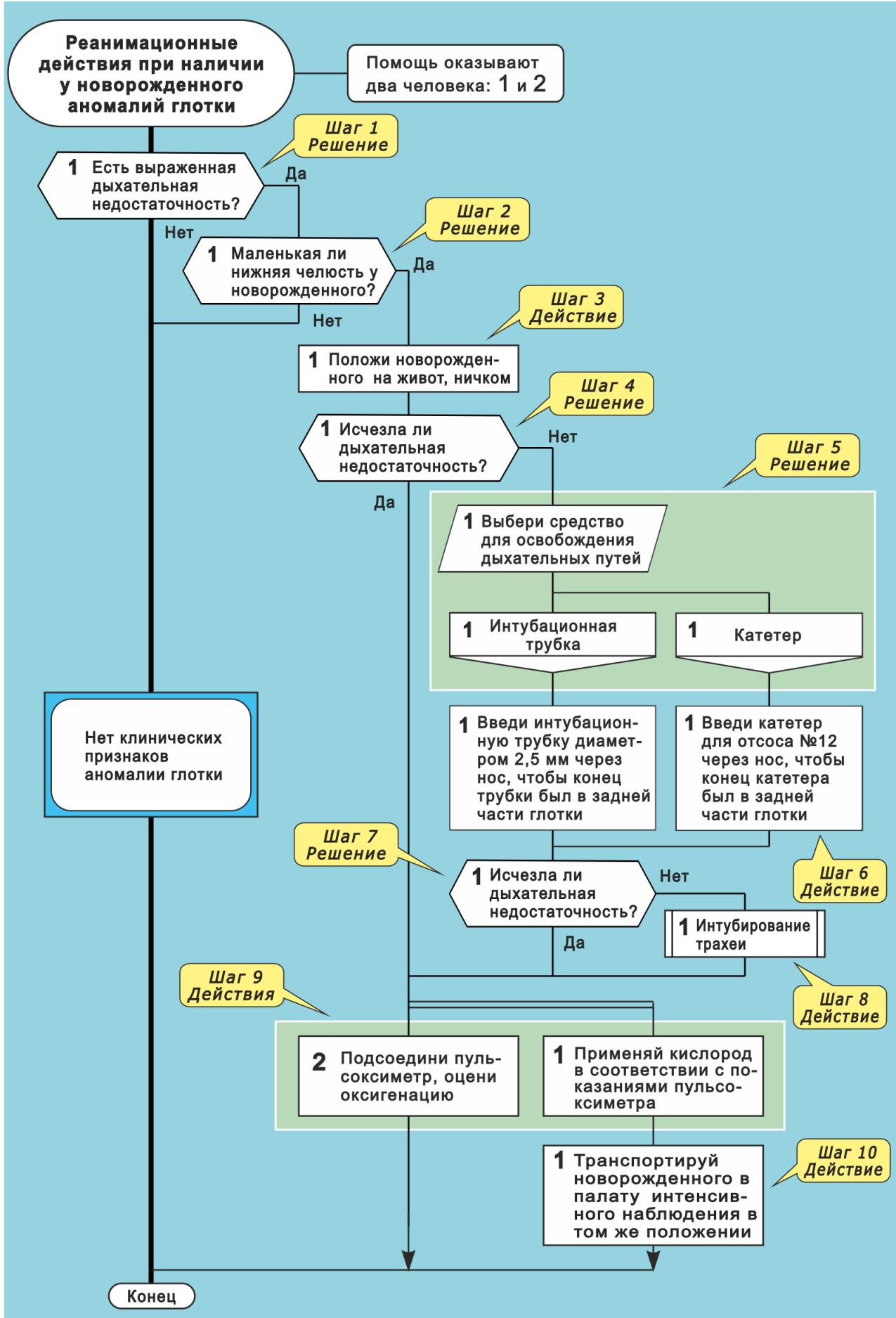


Рис. 4. Самый длинный маршрут алгоритма содержит 10 шагов

## **ПОВТОРЕТЬ МАТЬ УЧЕНЬЯ. ТЕЗИСЫ ДЛЯ ЗАПОМИНАНИЯ**

- Профессиональный медицинский язык — это язык медицинской литературы.
- Нынешний профессиональный медицинский язык представляет опасность для пациентов. Он порождает ошибочные действия медицинского персонала, которые могут повлечь за собой смерть, инвалидность или причинить иной вред больному.
- Подавляющее большинство медицинских алгоритмов представлено в мировой медицинской литературе в виде текста на естественном языке. Это недопустимо, потому что *обычный язык не пригоден для записи безошибочных алгоритмов*.
- Широко распространенное использование медицинского языка для записи медицинских алгоритмов является некорректным (безграмотным) и подлежащим исправлению. В этом заключается одна из причин врачебных ошибок.
- Понятие алгоритма вступает в противоречие с идеалами и нормами сложившихся представлений о медицинской науке, с историческими традициями и неформальным стилем медицинской литературы.
- Обострившаяся ситуация с врачебными ошибками и проблема безопасности пациентов вынуждают обратить особое внимание на это противоречие и устраниить его.
- Необходимо осуществить реформу профессионального медицинского языка (языка медицинской литературы, учебников, стандартов, руководств, клинических рекомендаций, протоколов), дополнив его медицинским алгоритмическим языком высокой точности — языком ДРАКОН.

## **ВЫВОДЫ**

1. Медицинские алгоритмы — это любые медицинские действия и решения, а также составленные из них сложные и разветвленные цепочки операций, выполняемые при профилактике, диагностике, лечении, экстренной помощи, реанимации, реабилитации, прогнозе.

*Уточнение.* Алгоритмами называются не сами действия и решения, а их точные описания.
2. Сегодня в мировой медицинской литературе даются неточные, приблизительные, неудовлетворительные описания алгоритмов, выполняемые с помощью естественного языка и неквалифицированных рисунков с многочисленными грубейшими нарушениями алгоритмических и эргономических правил.
3. *Медицинские алгоритмы высокой точности* обладают несомненным преимуществом по сравнению с ныне существующими и позволяют устраниить недостатки.
4. Актуальной задачей является алгоритмизация медицинской литературы, широкое использование *медицинских алгоритмов высокой точности* в медицинских учебниках, стандартах, руководствах, клинических рекомендациях, протоколах.
5. Правила создания и внешний вид медицинских алгоритмов высокой точности однозначно заданы и детально описаны в визуальном медицинском алгоритмическом языке высокой точности — языке ДРАКОН.
6. Язык медицинской литературы — это естественный текстовый язык. Как и любой естественный язык, он не обеспечивает необходимую точность изложения сложных вопросов. Он не приспособлен для описания медицинских алгоритмов, потому что для записи алгоритмов нужен специальный строгий язык.
7. Чтобы ликвидировать «алгоритмическую отсталость» медицины, необходимо:

- принять медицинские алгоритмы высокой точности в качестве стандарта для медицинской литературы,
- принять визуальный медицинский алгоритмический язык высокой точности (язык ДРАКОН) в качестве медицинского стандарта,
- изменить методику преподавания медицинских дисциплин в медицинских университетах и системе последипломного образования, опираясь на алгоритмы высокой точности,
- использовать медицинские алгоритмы высокой точности в медицинских экспертных системах (данная задача выходит за рамки книги).

## Глава 2

# ПОЧЕМУ МЕДИЦИНА УБИВАЕТ И КАЛЕЧИТ ПАЦИЕНТОВ?

### ЧТО ТАКОЕ МЕДИЦИНСКАЯ ОШИБКА?

Жила-была девушка на берегу океана в городе Бостон. Ее звали Бетси Леман. Случилось так, что она попала в больницу. И вдруг...

Почему? Почему очаровательная Бетси, обозреватель газеты Бостон Глоб так нелепо погибла? Потому что произошла медицинская ошибка — смертельная передозировка при химиотерапии [27].

Вилли Кингу тоже не повезло. Ему надо было ампутировать ногу. Операция прошла успешно. Но... с ошибкой. Надо было удалить больную ногу, а здоровую оставить. Увы, карта легла по-другому [27].

Бывает ли такое в России? Да, бывает. У нас есть свой, русский Вилли Кинг. Бедняге нужно было удалить разрушенное раком правое легкое. Но Господь опять не досмотрел — вырезали здоровое, оставив на память пораженное опухолью. Вот такая пневмонэктомия [28].

Единичные случаи? Исключения из правил? Если бы так. На самом деле, это широко распространенное зло, имеющее характер всемирной эпидемии. Врачебные ошибки — сложное, недостаточно изученное и очень опасное явление.

Что говорят  
эксперты

«Несмотря на растущее техническое оснащение медицинских учреждений, повышение квалификации врачей и успехи медицинской науки, количество больных, пострадавших от дефектов медицинской помощи, во всех странах нарастает» [178]

## **НЕ ЛГАТЬ, НЕ ИЗВОРАЧИВАТЬСЯ...**

Гиппократ рассматривал ошибки как источник знаний: «Если мы будем требовательны к себе, то не только успех, но и ошибка станет источником знания» [29].

Основоположник военно-полевой хирургии, анатом Николай Пирогов (1810–1881) призывал сразу сообщать о неудачах, чтобы предостеречь коллег:

«Я считаю священной обязанностью немедленно обнародовать свои ошибки и их последствия для предупреждения и назидания другим, еще менее опытным, от подобных заблуждений» [30].

Еще один знаменитый врач, пионер абдоминальной хирургии Теодор Бильрот (1829–1894) часто повторял:

«Только слабые духом, хвастливые болтуны и утомленные жизнью боятся открыто высказываться о совершенных ими ошибках. Кто чувствует в себе силу сделать лучше, тот не испытывает страха перед признанием своей ошибки» [31].

— Ничто так не способствует врачебной спеси, как игнорирование или забвение собственных ошибок, — напоминает хирург Николай Петров [30].

А вот что думает Жан-Луи Пти: «Ошибки являются только ошибками, когда у тебя есть мужество их обнародовать, но они становятся преступлением, когда гордыня тебя побуждает их скрыть» [32].

— Врач должен иметь мужество сознаться в своих ошибках, не лгать, не изворачиваться, — настаивает невропатолог Хаим-Бер Ходос [32].

## **СЛОМАННЫЕ СУДЬБЫ И ЗАГУБЛЕННЫЕ ДУШИ. ЗОЛОТОЙ ФОНД МЕДИЦИНЫ**

Однако не все были столь мудрыми. Медицинские ошибки зачастую скрывались от общественности, отрицались и замалчивались [33]. Тысячи людей обращались к врачам в надежде на исцеление, но вместо помощи получали путевку на кладбище.

На протяжении тысячелетий ошибки и неудачи были Суровым Учителем не только для начинающих врачей, но и для лучших целителей мира. Бесконечный конвейер сломанных судеб и загубленных душ терялся в волнах вечности и плавно превращался в драгоценные крупицы медицинского опыта.

«Только через грустный опыт отстаивается золотой фонд медицины», — замечает кардиохирург Николай Амосов [34]. Ему вторит американский врач Ной Фабрикант: «Иные врачи двадцать лет кряду делают одни и те же ошибки и называют это клиническим опытом» [35].

Проблема ошибок — одна из труднейших в медицине. К сожалению, она по-прежнему остается заброшенной нищенкой на роскошном празднике современной науки.

Врачебные ошибки были, есть и, по-видимому, будут всегда, пока врачеванием занимается человек. Сегодня важно на научной основе разработать эффективные меры, позволяющие свести ошибки к минимуму.

## **СЕНСАЦИОННЫЙ ДОКЛАД: СКОЛЬКО ЛЮДЕЙ БЕЗВИННО ГИБНЕТ В БОЛЬНИЦАХ США?**

Первый шаг к научному пониманию проблемы был сделан в Национальной академии наук США, медицинское отделение которой носит название *Институт*

медицины (Institute of Medicine) [36]. В 2000 году, на основании тщательных исследований, Институт опубликовал доклад под интригующим названием «Человеку свойственно ошибаться» с подзаголовком «Создание более безопасной системы здравоохранения» (To Err is Human: Building a Safer Health System) [1]. 300-страничный труд сразу же привлек к себе внимание и превратился в сенсацию.

Поражают и масштабы проблемы, и научная честность авторов. Они не стали отстаивать честь мундира, не побоялись выносить сор из избы. Оказывается, медицинские ошибки в больницах США являются причиной смерти от 44 000 до 98 000 человек в год [37]. Цифры озадачивают и настороживают. Они означают, что «в американских больницах каждые полгода погибает больше американцев, чем за всю Вьетнамскую войну» [38].

Авторы признают, что врачебные ошибки занимают одно из ведущих мест в структуре смертности населения США. Даже если взять нижнюю оценку (44 000 человек), смертность из-за врачебных ошибок превышает значение восьмой ведущей причины смерти в США. По вине врачей умирает больше людей, чем от дорожно-транспортных происшествий (43 458 жертв), от рака молочной железы (42 297 жертв), от СПИДа (16 516 жертв)<sup>4</sup> [39].

В докладе делается вывод, что больница гораздо опаснее самолета. Потому что риск смерти вследствие врачебной ошибки намного больше, чем риск гибели в авиационной аварии («risk of dying as a result of a medical error far surpasses the risk of dying in an airline accident») [40].

## БОЛЬНИЦА В 10 000 РАЗ СТРАШНЕЕ АВИАКАТАСТРОФЫ

Вскоре выяснилось, что сходная картина имеет место не только в Америке, но и в других регионах мира — в Азии, Африке и др. [41] [42].

В материалах Всемирной организации здравоохранения говорится, что пребывание в больнице в 10 000 раз опаснее, чем полет на самолете:

«Вероятность несчастного случая в самолете составляет 1 на 3 миллиона. Риск возникновения несчастного случая в больнице 1 на 300» [43].

Эти данные послужили основанием для развертывания широкой кампании сначала в США, а затем и на международном уровне, по борьбе с ошибками в медицине [1] [44] [45] [46] [47] [2] [48].

## РАСПРОСТРАНЕНИЕ ИДЕИ

В последнее время в разных странах мира предпринимаются энергичные усилия по предотвращению медицинских ошибок. Эти усилия чрезвычайно важны и крайне необходимы. По рекомендациям Института медицины [1] [2] в конгрессе США были проведены слушания и приняты законы о безопасности пациентов, подписанные президентами Биллом Клинтоном [48] и Джорджем Бушем младшим [4, 5].

В мировой системе здравоохранения выявлены и частично устранены многие неблагоприятные ситуации, представляющие опасность для пациентов. Вместе с тем, полезно помнить, что чаще всего выявляются и устраняются относительно простые ошибки. Например, такие, как внутрибольничные инфекции (hospital-acquired infections), поражающие каждый год 1,4 миллиона человек [49], ошибки при переливании крови и т.д.

В целом, проблема по-прежнему далека от разрешения [50] [51].

<sup>4</sup> Все цифры согласно статистике за 1997 год.

# **ОШИБКИ МЫШЛЕНИЯ ВРАЧЕЙ. МЕДИЦИНСКИЙ ЯЗЫК КАК ИСТОЧНИК НЕПРИЯТНОСТЕЙ**

Медицинские ошибки зависят от многих причин, в том числе, от недостатков медицинского языка. К сожалению, эта последняя причина обычно не учитывается. А зря. На самом деле, она очень важна.

Дефекты медицинского языка могут оказывать негативное и даже разрушительное воздействие на профессиональное мышление врачей. Приходится признать, что предпринимаемые сегодня меры, предложенные в докладах Института медицины, являются недостаточными, поскольку они почти не затрагивают ошибки мышления [52].

Между тем, интеллектуальные ошибки врачей, то есть ошибки мыслительных операций, представляют наибольший интерес, так как свидетельствуют об исключительной сложности медицинского мышления. Они меньше всего исследованы, тесно связаны с языком и нуждаются в тщательном анализе.

В книге, которую вы держите в руках, проводится мысль, что медицинский язык имеет важный дефект, представляет опасность для пациентов и потому нуждается в серьезной доработке и совершенствовании.

## **ЧРЕЗМЕРНАЯ СЛОЖНОСТЬ МЫШЛЕНИЯ**

По нашему мнению, на первый план выступает новая проблема — *проблема сложности медицинского мышления*.

Известно, что медицина чрезвычайно сложна и с каждым годом продолжает усложняться. Бурное развитие медицинских знаний и технологий предъявляет все новые и новые требования к квалификации медперсонала и медицинскому образованию.

Как следствие, увеличивается нагрузка на мозг врачей, вынуждая их решать все более сложные мыслительные задачи. В том числе, чрезмерно сложные. При этом забывают, что врач живой человек, а его психофизиологические характеристики и способности отнюдь не безграничны.

Чрезмерная сложность мышления врачей вызывает обоснованную тревогу, так как именно она часто является причиной медицинских ошибок с тяжелыми последствиями.

В книге описан когнитивный метод, позволяющий устраниить чрезмерную сложность мышления, для того чтобы помочь врачам и облегчить их жизнь. Метод дает возможность упростить чрезмерно сложную лечебно-диагностическую задачу и превратить ее в легко обозримую, которая не вызывает затруднений у врача.

## **ОШИБОЧНЫЙ ДИАГНОЗ**

Что означает медицинская ошибка при диагностике, приведшая к смерти или инвалидности больного? Она означает, что нагрузка на мозг лечащего врача стала чрезмерной и непосильной.

В результате запредельной нагрузки, мозг врача прекратил правильно функционировать, так как мыслительная задача оказалась ему не по силам. Она оказалась слишком трудной, превышающей критический порог. Если бы нагрузка не была чрезмерной, врач бы не ошибся, и пациент не пострадал бы.

Кто же виноват в нанесении ущерба пациенту? Врач, который установил неверный диагноз? Или, может быть, виновато общество, которое ставит перед врачом непосильные для его мышления задачи, жестко ограничивая время на их решение?

Мнение  
экспертов

«В среднем более чем у 15% больных заключительные клинические диагнозы, как показывают аутопсии, являются ошибочными... Основное заболевание при жизни пациента не распознается клиницистом примерно у 15 — 20% больных, умерших в стационарах нашей страны. Близкие цифры характерны и для стационаров других стран» [178].

## ИНТЕЛЛЕКТУАЛЬНАЯ ПЕРЕГРУЗКА ВРАЧЕЙ. ДОПУСТИМА ЛИ ОНА?

Общество должно помочь врачам, облегчить их тяжелый интеллектуальный труд, сделать его менее сложным и напряженным. Необходимо уважать достоинство врача. Поэтому труд должен быть *посильным*.

Общественность должна знать, что большинство врачей — отнюдь не гении, они не обладают выдающимися способностями и талантами. Это рядовые лечащие врачи больниц, поликлиник и медицинских центров, имеющие обычные, средние способности.

Интеллектуальная нагрузка на их мозг должна быть соразмерной их силам и способностям. Чрезмерная сложность мышления должна быть исключена с помощью научно-обоснованных методов.

## ПОЧЕМУ ИСКАЗИЛИ КЛЯТВУ ГИППОКРАТА?

Две с половиной тысячи лет назад, Гиппократ написал в своей Клятве: нужно лечить «сообразно с моими силами и моим разумением» [53]. Однако в современной Клятве врача (Женевской декларации) и Клятве, на которой присягают выпускники медицинских институтов, эти важные слова почему-то вычеркнули [54]. Правильно ли это?

Интеллектуальная перегрузка врачей («сверх моих сил и за пределами моего разумения») недопустима.

Почему? Потому что она порождает медицинские ошибки, влекущие за собой смерть, инвалидность или иной ущерб для пациентов.

Является ли эта проблема социально значимой? Несомненно. Опираясь на научный метод и идеи когнитивной эргономики, можно сделать вывод: *во имя безопасности пациентов необходимо превратить непосильную для врача интеллектуальную работу (порождающую врачебные ошибки) в посильную и, по возможности, безошибочную*.

Когнитивная  
эргоноомика

Это эргономика познавательных и мыслительных процессов

## НАУКА О ЧЕЛОВЕЧЕСКИХ ОШИБКАХ

Научной основой для решения проблемы является эргономика, которая учит, как нужно бороться с человеческими ошибками в различных, прежде всего, технических областях деятельности человека.

Вспомним, что в середине XX века — в процессе бурного развития техники и создания систем «Человек–Машина» — многочисленные ошибки операторов сложных технических систем (пилотов военной и гражданской авиации, операторов атомных электростанций, химических реакторов, космонавтов) приводили к тяжелым и нередко трагическим последствиям.

Анализ этих ошибок породил эргономику и коренным образом изменил форму представления информации в системах «Человек — Техника». Это повысило наглядность сообщений, снизило чрезмерную нагрузку на человека-оператора и облегчило условия его работы. В итоге число ошибок резко сократилось.

Есть ли аналогия между ошибками операторов и ошибками врачей? Можно ли использовать новые формы представления информации, чтобы обеспечить безопасность пациентов? Да, можно.

Ниже показано, что эргономика — это хорошая идея для борьбы с медицинскими ошибками. Речь пойдет о применении эргономики для совершенствования медицинской литературы.

Медицинская  
когнитивная  
эргономика

Это ветвь эргономики, призванная усовершенствовать формы представления знаний в медицинской литературе, чтобы студенты и врачи могли приобретать медицинские знания быстрее (за меньшее время), легче (с меньшими усилиями) и глубже в процессе зрительного восприятия, осмыслиения, понимания и усвоения содержания медицинских учебников, руководств, стандартов, клинических рекомендаций и протоколов, используя более совершенные познавательные и мыслительные процессы

## ПУТЕШЕСТВИЕ В ШАПКЕ-НЕВИДИМКЕ. ЭССЕ ОБ ЭРГОНОМИКЕ

Инженерные психологи и эргономисты ввели важное понятие — Средства Отображения Информации (СОИ) [55] [56] [57]. Что это такое?

Наденем шапку-невидимку и, ловко проскользнув мимо охранников, проникнем в святая-святых — в зал управления ядерной электростанции. Понааблюдаем за работой здешнего гуру — оператора реакторного отделения [58]. Он священнодействует за пультом управления, на котором размещаются различные экраны, табло, транспаранты, мнемосхемы, сигнальные лампочки. Это и есть средства СОИ — глаза и уши оператора. Они предъявляют ему информацию о работе ядерного реактора в наиболее удобном виде.

Оператор должен хорошо знать, что творится внутри реактора. Если он, не дай Бог, чего-то не поймет, то может по ошибке выполнить неправильные действия, нажать не ту кнопку. Такая оплошность может привести к нежелательным и опасным последствиям.

Чтобы исключить ошибки, на атомной станции предпринимаются особые, уникальные меры. Надо не просто снабдить оператора надежной информацией. Этого

мало. Нужно гарантировать полное и точное понимание человеком текущей обстановки.

В реакторном отделении протекают физические процессы умопомрачительной сложности (еще хуже, чем в медицине). Но информация об этих процессах, предъявляемая оператору, должна быть простой и предельно ясной.

Она должна исключать всякую возможность кривотолков. Поэтому средства СОИ проектируют по очень строгим правилам. Указанные правила разрабатываются на основании глубоких и тонких инженерно-психологических экспериментов [58] [59] [60] [61] [62] [63].

Информация, которую получает оператор, должна:

- отображать реальную ситуацию в ядерном реакторе;
- учитывать закономерности и характеристики человеческого восприятия, памяти, мышления;
- быть наглядной, легкой для усвоения;
- соответствовать задачам трудового процесса по управлению реактором;
- соответствовать возможностям человека по приему информации.

Информацию, удовлетворяющую перечисленным требованиям, можно назвать эргономичной. Такая информация позволяет оператору уяснить суть проблемной ситуации быстро, без трудоемкого анализа.



**Рис. 4а.** Средства отображения информации на ядерной электростанции в Пакше (Венгрия)

## НЕ ЗАСТАВЛЯЙТЕ МЕНЯ ДУМАТЬ!

Оператор вправе сказать: «Не заставляйте меня думать!» над второстепенными вопросами, которые отвлекают меня от главного. Это значит, что замысловатые и мудреные сообщения, которые требуют мучительных раздумий, должны быть полностью исключены. Сложная информация должна подаваться человеку в ясной и доходчивой форме, в заранее разжеванном виде, не требующем мысленного перекодирования на «свой» язык [59].

Благодаря этому, оператор получает информацию, которая обладает поистине бесценными свойствами. Она вносит ясность, всегда помогает, никогда не запутывает. Она облегчает умственную деятельность, позволяет устраниить ошибки понимания, обеспечивает интеллектуальный комфорт. То есть создает для человека почти идеальные условия работы.

На основании теоретического анализа и обширного практического опыта специалисты по эргономике разработали правила, позволяющие разумно выбирать формы представления информации для человека-оператора [58] [59] [60] [61] [62] [63] [64].

Повторим  
важную мысль

Благодаря применению эргономических (инженерно-психологических) методов в системах «Человек — Техника»:

- коренным образом изменились и улучшились формы представления информации для человека-оператора,
- чрезмерная сложность мышления человека была полностью исключена,
- работа оператора стала удобной и комфортной,
- число ошибок резко сократилось.

## КАК УЛУЧШИТЬ РАБОТУ УМА ВРАЧЕЙ

В медицине следует поступить точно так же — коренным образом изменить и улучшить форму представления знаний в медицинских учебниках, стандартах, руководствах, клинических рекомендациях, протоколах.

**Что значит «изменить и улучшить форму представления знаний»?**

Известно, что «любой язык есть форма представления всех тех знаний, которые могут быть представлены на этом языке» [65]. Отсюда следует, что медицинский язык есть форма представления медицинских знаний. Таким образом, изменить и улучшить форму представления медицинских знаний — значит *изменить, усовершенствовать и улучшить медицинский язык*.

**Что значит «усовершенствовать язык»?**

Можно ли его усовершенствовать? Или это в принципе невозможно?

В свое время Эмиль Бенвенист писал: «Невозможно вообразить человека без языка и изобретающего языка» [66].

С тех пор утекло много воды. В мире накоплен огромный опыт создания искусственных языков. Автор этих строк в течение многих лет разрабатывал и совершенствовал языки для ракетно-космической отрасли. Что касается естественного языка, полезно изучить книгу Рубена Будагова «Что такое развитие и совершенствование языка?» [67].

Нет сомнения, что научно обоснованное усовершенствование медицинского языка позволит уменьшить чрезмерную интеллектуальную нагрузку на врача, сделать ее посильной и комфортной. И за счет этого существенно сократить количество врачебных ошибок [52].

Разумеется, формы представления знаний и информации для разных областей могут отличаться, но общий принцип — принцип эргономизации знаний — остается неизменным [68].

Специалист по медицинским алгоритмам, профессор Владимир Тавровский указывает: чтобы устраниТЬ противоречие между информационным взрывом, в условиях которого работает врач, и его ограниченными возможностями, «клинический опыт и опыт управления надо уложить в строгие алгоритмы поведения» [69]. Золотые слова!

Как уменьшить погрешности мышления? Какие методы когнитивной эргономики лучше использовать на практике? С ответами на эти вопросы можно познакомиться в книгах [70] [71] [72] [73] [74].

## ЭРГОНОМИКА + АЛГОРИТМЫ: ВЫСТРЕЛ ДУПЛЕТОМ

Цель книги — облегчить жизнь врача и защитить его от неудач. В чем изюминка нашего метода? Тут не одна, а две изюминки. Первая — эргономика, вторая — алгоритмы.

Хитрость в том, что применять их надо не порознь, а вместе — получится взрывчатая смесь под названием *эргономичные алгоритмы*. Взрывной эффект состоит в том, что произойдет значительное облегчение работы врача.

Эргономичные алгоритмы имеют огромные преимущества и отличаются от обычных, как небо от земли. Обычные алгоритмы трудны для понимания (черт ногу сломит), а эргономичные, наоборот, воспринимаются легко и с удовольствием.

Заветная  
мысль автора

Необходимо одновременно осуществить и эргономизацию,  
и алгоритмизацию медицинской литературы [52]

## ВЫВОДЫ

1. Врачебные ошибки — сложное, недостаточно изученное и очень опасное явление. Ошибки могут повлечь за собой смерть, стойкую инвалидность или иной ущерб для пациентов.
2. Необходимо на научной основе разработать эффективные меры, позволяющие свести ошибки к минимуму.
3. Важный шаг к решению проблемы ошибок и безопасности пациентов сделан в Национальной академии наук, техники и медицины США — в трех докладах Института медицины [1] [2] [3], опубликованных в 2000, 2001 и 2015 годах.
4. Врачебные ошибки занимают одно из ведущих мест в структуре смертности населения США.
5. По рекомендациям Института медицины в конгрессе США были проведены слушания и приняты законы о безопасности пациентов.
6. Выводы и предложения Института медицины США являются важными, но недостаточными. Они не учитывают тот факт, что профессиональный медицинский язык имеет серьезный дефект и представляет опасность для пациентов.
7. Медицинские ошибки зависят от многих причин, в том числе, от недостатков медицинского языка.
8. Дефекты медицинского языка могут оказывать негативное и даже разрушительное воздействие на профессиональное мышление врачей, что является причиной ошибок.
9. Опираясь на достижения когнитивной эргономики и теории эргономичных алгоритмов, необходимо коренным образом изменить и улучшить форму представления знаний в медицинских учебниках, стандартах, руководствах, клинических рекомендациях, протоколах.
10. Научно обоснованное усовершенствование медицинского языка позволит уменьшить чрезмерную интеллектуальную нагрузку на врача, сделать ее посильной и комфортной. И за счет этого существенно сократить количество врачебных ошибок.

*Рекомендация.* Пропустите следующую главу и сразу приступайте к главе 4.

## Глава 3

# АХИЛЛЕСОВА ПЯТА МЕДИЦИНЫ И ПРОБЛЕМА НЕОПРЕДЕЛЕННОСТИ

### НЕ ЧИТАЙТЕ ЭТУ ГЛАВУ!

В главе обсуждаются вопросы повышенной трудности:

- Кто хозяин понятия *алгоритм*: математики или медики?
- Есть ли принципиальное отличие между медицинскими и математическими алгоритмами? Если есть, в чем конкретно оно выражается?
- Возможен ли компромисс в споре медиков и математиков?
- В чем состоит проблема «неопределенности» в медицинской литературе?
- Является ли неопределенность источником врачебных ошибок?
- Можно ли снизить или устранить неопределенность с помощью «медицинских алгоритмов высокой точности»?
- Должны ли руководители здравоохранения создать условия для решения проблемы неопределенности?

Подобные вопросы интересуют далеко не всех. Если вам не интересно, главу можно пропустить.

### НА СВОЙ СТРАХ И РИСК

Что такое алгоритм? Это строгое и чрезвычайно важное математическое понятие. Оно вступает в острое противоречие со свободным, творческим, неформальным стилем медицинского мышления. Стилем, который формировался в процессе длительного исторического развития в течение тысячелетий со времен Гиппократа.

Алгоритм — символ безукоризненной математической строгости и точности. Такой точности (алгоритмической точности клинического мышления) медицина никогда не знала. Алгоритмическая точность является «фантастической», немыслимой для врачей.

Вместе с тем, наиболее дальновидные представители медицинского сообщества интуитивно чувствовали, что в понятии алгоритма таится гигантская мощь, потрясающая взрывчатая сила, которая со временем может принести здравоохранению огромные выгоды.

Они на свой страх и риск начали применять слово *алгоритм* в медицинской литературе как полезную метафору, действуя скорее по наитию, как алхимики и

заговорщики, а не как ученые. Формально это было неправильно, потому что медицинский алгоритм не имеет почти ничего общего с настоящим математическим алгоритмом.

Подчеркнем: никто и никогда не пытался обосновать и доказать возможность и необходимость введения понятия «алгоритм» в медицину. Это было чисто интуитивное прозрение энтузиастов, которое ворвалось и воплотилось в жизнь незаконно, стихийно, явочным путем, как нелегальные мигранты в современную Европу.

## СМЕНА ЭПОХ В МЕДИЦИНЕ

Хотя формально использование нового понятия было неправильным, но в стратегическом смысле это была мудрая и многообещающая инициатива. Нет сомнения, что она знаменует прорыв в медицинской науке — прорыв, который пока еще не состоялся, но который, по-видимому, произойдет в скором будущем.

Инициатива не пропала даром. В медицинских трудах, руководствах и образовательных стандартах метафора (или, если угодно, термин) *алгоритм* получил прописку, укоренился, пустил ростки и молодые побеги. Он стал успешно размножаться, почковаться и часто применяться.

Похоже на то, что в медицине происходят скрытые тектонические процессы огромной важности. Мы, по-видимому, являемся свидетелями смены эпох. Заканчивается эпоха полифонии, непримиримых разногласий медицинских школ и приближительности. Начинается эпоха международной стандартизации, внедрения методов высокой точности и доказательной медицины.

## ПРОТИВОРЕЧИЕ МЕЖДУ ВРАЧАМИ И МАТЕМАТИКАМИ. ПОДДЕЛКА ПОД АЛГОРИТМ

Ясно, однако, что противоречие никуда не исчезло. Медицинский алгоритм по-прежнему был и остается неточным и приблизительным. Это никакой не алгоритм, а всего лишь суррогат алгоритма. Полезный и необходимый, но все-таки суррогат. По мнению математиков, его никак нельзя считать настоящим алгоритмом.

Можно ли устранить это поистине серьезное противоречие между врачами и математиками? Можно ли сблизить позиции двух древнейших и столь нужных человечеству наук (медицины и математики), чтобы найти разумный компромисс? Можно ли решить спор, возникший из-за понятия алгоритм, путем беспристрастных логических рассуждений?

Да, это возможно, если обе стороны готовы пойти на уступки. Попробуем разобраться.

## НЕУДАЧНОЕ И НЕПРИЕМЛЕМОЕ ИСПОЛЬЗОВАНИЕ ТЕРМИНА «АЛГОРИТМ» В МЕДИЦИНЕ

Понятие *алгоритм* в медицинском мире не имеет строгого определения, допускает различные толкования и графические представления. Стандартизация отсутствует. Господствует разноголосица. Кто в лес, кто по дрова [75] [76] [77] [78] [79] [80] [81].

Начнем со случаев, где термин *алгоритм* употребляется правильно или почти правильно:

— в медицинском программировании;

— в автоматических медицинских вычислениях по формуле. Вычислительные услуги такого рода в режиме онлайн оказывает, например, коммерческая фирма «Медицинские алгоритмы» MEDAL (The Medical Algorithms Company). Она продает врачам доступ к программам, которые именуются «медицинскими алгоритмами». В базе данных фирмы хранится двадцать тысяч алгоритмов, которыми она торгует [82].

Оба эти случая нас не интересуют и далее не рассматриваются, так как они относятся к медицинской информатике, которая выходит за рамки книги.

Вернемся к нашей основной теме — анализу медицинской литературы. Не будет преувеличением сказать, что врачи во многих случаях используют термин *алгоритм* некорректно, скорее, как дань моде, не до конца понимая его глубинный смысл.

Главный недочет большинства медицинских трактовок понятия *алгоритм* — отсутствие необходимой точности, приблизительность, недосказанность, смысловые лакуны, отсутствие нужных уточнений, отсутствие важных деталей, подробностей и условий

В медицинских учебниках, стандартах, руководствах и пр. используются неточные, приблизительные, неудовлетворительные описания медицинских алгоритмов. Подобные описания нарушают элементарные алгоритмические и эргономические правила, провоцируют врачебные ошибки и опасны для пациентов.

Все эти погрешности зачастую исключают возможность воспроизвести алгоритм, опираясь на его описание. Это значит, что нельзя воспроизвести (повторить) процессы диагностики и лечения, используя только алгоритмы. Потому что указанные процессы описаны в алгоритмах далеко не полностью.

## ПРОПАСТЬ МЕЖДУ НАСТОЯЩИМ И МЕДИЦИНСКИМ АЛГОРИТМОМ

Настоящий математический алгоритм можно воспроизводить (повторять) сколько угодно раз с абсолютной точностью. Возьмем для примера детский алгоритм сложения столбиком.

$$\begin{array}{r} 37 \\ + \quad 21 \\ \hline 58 \end{array}$$

Выучив правила сложения в начальной школе, человек приобретает умение складывать числа с нужной точностью. Почему?

Алгоритм сложения столбиком содержит в себе все указания, необходимые для этого. По этой причине алгоритм сложения может выполнять кто угодно, даже электронный робот, калькулятор или компьютер.

Медицинский алгоритм устроен иначе. Он содержит в себе лишь часть указаний, а остальные подразумевает (утаивает), рассчитывая на догадливость живого человека.

Поэтому *алгоритм ведения пациента при нормотензивной глаукоме* нельзя возложить на робота. Для этого непременно нужен врач, профессиональный офтальмолог.

## МОЖНО ЛИ ИСПРАВИТЬ ПОЛОЖЕНИЕ И УВЕЛИЧИТЬ ТОЧНОСТЬ МЕДИЦИНСКИХ АЛГОРИТМОВ?

Да, можно. Однако, чтобы коренным образом поправить дело и достичь желаемого эффекта, точность (определенность) алгоритмов следует повысить не чуть-чуть, а кардинально.

Это очень жесткое требование. Алгоритмы, представленные в мировой медицинской литературе, не удовлетворяют требованию точности, за редким исключением. Врачи не умеют описывать точные алгоритмы и не понимают, что это такое. Подчеркнем: точность алгоритмов — совершенно новое, неслыханное, беспрецедентное для врачей требование.

Согласно развивающейся идеи, прежние (неточные, не полностью определенные) описания медицинских алгоритмов должны со временем отмереть, навсегда сойти со сцены. И уступить место принципиально новому типу алгоритмов — *медицинским алгоритмам высокой точности*.

Здесь имеется в виду, что *точность алгоритма* — синоним термина «определенность (детерминированность) алгоритма».

## НЕОБХОДИМО ИСПОЛЬЗОВАТЬ ПРАВИЛЬНУЮ ТЕРМИНОЛОГИЮ

С математической точки зрения, медицинский алгоритм — это совсем не алгоритм. А что же это такое? Это всего лишь *алгоритмическое предписание* (algorithmic prescription), то есть (по сравнению с алгоритмом) вещь ущербная, неполноценная, второсортная.

Между алгоритмом и алгоритмическим предписанием лежит глубокая пропасть, маракотова бездна. Наша задача — построить мост через эту пропасть, мост взаимопонимания между медиками и математиками.

Прежде всего, следует внести ясность в терминологию. Мы будем опираться на тезис, который сформулировал математик Николай Непейвода:

«Необходимо различать алгоритм и алгоритмическое предписание, имеющее внешнюю форму алгоритма, но включающее не до конца определенные шаги» [180].

На этот счет существует обширная литература, восходящая к пионерской работе Льва Ланды «Алгоритмизация в обучении» [83]. Ланда первым указал различие, но мы предпочитаем более точную формулировку Непейводы<sup>5</sup>.

Тезис Ланды-Непейводы создает научный фундамент для четкого разграничения медицинских и математических алгоритмов.

## ЧТО ТАКОЕ «ОПРЕДЕЛЕННОСТЬ» АЛГОРИТМА?

Определенность говорит о том, что каждое указание алгоритма должно быть точным, четким, однозначным и не оставлять места для случайных, произвольных толкований и действий.

<sup>5</sup> Приведем исходную формулировку Льва Ланды: «В отличие от алгоритмов в строгом математическом смысле, алгоритмические предписания допускают правила, которые обращены не только к формальным, но и к содержательным операциям... Понятие предписания алгоритмического типа является менее точным (в математическом смысле), чем понятие алгоритма» [183].

В тезисе Непейводы есть фраза «не до конца определенные шаги». Что она означает?

Смысл в том, что в алгоритмическом предписании (читай — в ущербном, неполноценном алгоритме) отсутствует надлежащий порядок. Некоторые правила описания и выполнения шагов алгоритма или не указаны, или неизвестны, или неудобны, или непонятны, или не соблюдаются.

По сути дела, это игра без правил. Или с плохими, не до конца продуманными правилами.

Повторим еще раз: в алгоритмическом предписании (в отличие от истинного алгоритма) нет необходимой точности, нет однозначности. Поэтому читатели алгоритмического предписания испытывают трудности и могут понять его смысл неправильно, неоднозначно, превратно, и так и эдак.

## ДВА ПОУЧИТЕЛЬНЫХ ПРИМЕРА

*Пример 1.* На книжной полке много книг. Одна из них с красным переплетом. Рассмотрим просьбу, точнее команду, обращенную к случайному гостю:

— Достаньте красную книгу.

Эту команду можно выполнить абсолютно точно (конечно, если гость не будет упираться). В условии сказано: На полке среди множества книг только одна имеет красный переплет. Следовательно, команда представляет собою строгий алгоритм.

*Пример 2.* Изменим условия и добавим на полку еще одну красную книгу.

Снова звучит команда:

— Достаньте красную книгу.

В отличие от предыдущего случая, появилась *неопределенность*. Неясно, какую из двух красных книг следует взять. Разные люди, выполняя команду, могут предпочесть разные книги.

Наша команда имеет следующие характеристики.

1. Она точно определяет действие, которое нужно выполнить («Достаньте»).
2. Она точно определяет класс объектов, из которых следует сделать выбор (класс «книги на книжной полке»).
3. Она четко указывает критерий, который следует использовать при выборе искомого объекта («книга в красном переплете»).

В предыдущих пунктах точно и однозначно указаны три характеристики. Тем не менее, критерий не является достаточным для того, чтобы человек смог выбрать нужную книгу из двух книг, отвечающих критерию.

Таким образом, перечисленные характеристики не позволяют точно описать действие, ведущее кциальному результату. Команда «Достаньте красную книгу» содержит некоторую степень неопределенности. Поэтому результат действия может оказаться неправильным.

Степень неопределенности, пусть даже небольшая, является глубинным источником многих человеческих ошибок.

## ЧТО ЭТО ОЗНАЧАЕТ ПРИМЕНЯТЕЛЬНО К МЕДИЦИНЕ?

Содержание предыдущего параграфа взято из книги «Парадигмы инструктивного развития» (Instructional Development Paradigms) [84], в которой, в частности, акцентируется внимание на алгоритмическом, пошаговом (step-by-step) подходе к приобретению знаний. Мы убедились, что неточность и неопределенность являются опасными и в ответственных случаях их следует избегать.

Медицина, несомненно, является ответственным делом, поскольку врачебные ошибки чреваты гибелью больных или нанесением им существенного вреда. Поэтому степень неопределенности в медицинской литературе (учебниках, стандартах, руководствах, клинических рекомендациях, протоколах) должна быть минимальной.

Можно ли этого добиться? Можно ли минимизировать степень неопределенности в медицинских изданиях? (Имеются в виду лишь те случаи, где наука располагает необходимыми знаниями).

Это чрезвычайно сложная задача. Для ее решения нужно многое изменить. Однако, если мы хотим сократить врачебные ошибки не на словах, а на деле, необходимо серьезно задуматься над проблемой неопределенности.

## АХИЛЛЕСОВА ПЯТА МЕДИЦИНЫ

Неопределенность в медицинской литературе является глубинным источником многих врачебных ошибок. Образно говоря, неопределенность — ахиллесова пята медицинских публикаций.

Если мы согласны с Институтом медицины США (Institute of Medicine) в том, что *больницы не должны убивать людей*, значит, следует обратить пристальное внимание на проблему неопределенности.

Обострение ситуации с врачебными ошибками, настоятельное требование пациентов о безотлагательном обеспечении их безопасности (patient safety) подводит ученых и руководителей здравоохранения к необходимости принять решительные меры по устранению неопределенности в медицинской литературе.

Чтобы решить поставленную задачу и устраниТЬ неопределенность, необходимо использовать *медицинские алгоритмы высокой точности*.

### Пояснение 1

- Медицинский алгоритм — это не алгоритм, а всего лишь алгоритмическое предписание.
- Это объясняется тем, что медицинский алгоритм не обладает свойством определенности (однозначности), которое является обязательным для алгоритмов.

### Пояснение 2

- Задача состоит в том, чтобы кардинально повысить качество медицинских алгоритмов и в максимальной степени приблизить их свойства к образцу — к свойствам настоящих математических алгоритмов.
- Для решения этой задачи вводится понятие «медицинский алгоритм высокой точности».

## СЦИЛЛА И МЕДИЦИНСКОЕ ОБРАЗОВАНИЕ

Выше уже говорилось: Сцилла следит за формальной строгостью медицинских алгоритмов, а Харибда обеспечивает удобочитаемость — чтобы алгоритмы были удобны для врачей. Это диаметрально противоположные требования. Чем больше строгость, тем труднее читать и понимать алгоритмы. И наоборот.

Проблема в том, что наша воображаемая Сцилла предъявляет чрезмерные требования. Она настаивает на строгой нотации типа Арден-синтаксиса (Arden Syntax), который используется в медицинском стандарте HL7 (Health Level 7) для определения и распространения медицинских знаний [85] [86].

Нет слов, для специалистов по медицинской информатике и программистов Арден-синтаксис — хорошее решение. Но для врачей он совершенно не пригоден. Нельзя врачей превращать в программистов.

В греческой мифологии Сцилла — чудище с шестью головами, которыми она похитила шесть спутников Одиссея. Такой образ нам не подходит.

Поэтому заменим Сциллу на скромного часового с одной головой, который зорко следит за формальной строгостью алгоритмов, но, в отличие от безумной Сциллы, хорошо понимает, что для врачей арден-синтаксис — это каша с гвоздями.

По нашему мнению, недопустимо вводить арден-синтаксис (или что-то подобное) в систему медицинского образования. Для описания медицинских алгоритмов, которые будут изучать студенты-медики, нужен легкий и удобный язык.

## ТРЕБОВАНИЯ К ЯЗЫКУ

Отсюда вытекают требования к языку для описания медицинских алгоритмов высокой точности. Последние должны быть:

- привлекательными и удобными для врачей;
- дружелюбными (people-friendly) по отношению к врачам;
- иметь текст на родном языке (а не на языке программирования).

Эти требования являются приоритетными и должны выполняться при всех условиях.

После этого можно начать разговор о строгости. Цель состоит в том, чтобы по возможности приблизиться к идеалу — к строгости математических алгоритмов:

- строгость алгоритмов должна быть обеспечена, но она должна быть «невидимой» и не создавать лишних трудностей для врачей;
- строгость должна быть обеспечена не с помощью формул, а с помощью графики (см. главы 4 и 12).

Иными словами, математика должна быть спрятана в графике и благодаря этому стать «невидимой». Как это сделать?

Медицинские алгоритмы следует рисовать не вручную, а с помощью специального графического редактора, который не позволяет делать ошибки при разработке алгоритмов (см. главу 12).

## МЕДИЦИНСКИЙ АЛГОРИТМ КАК ФУНДАМЕНТАЛЬНОЕ ПОНЯТИЕ

До сих пор наши рассуждения шли по классической схеме и опирались на три тезиса.

1. Математический алгоритм — образец совершенства.
2. Медицинский алгоритм — ущербный, неполноценный, второсортный объект.
3. А раз так, нужно усовершенствовать его, равняясь на образец.

Отчасти это верно, но только отчасти.

Настало время взглянуть на проблему более глубоко и выделить главное. Суть в том, что медицинские алгоритмы обладают качественным своеобразием. От них зависит жизнь, здоровье и благополучие человечества. Это значит, что они не менее

важны, чем математические алгоритмы. Следовательно, тезис 2 является ошибочным и должен быть исключен.

Таким образом, медицинские алгоритмы нельзя считать ущербными, неполноценными и второсортными. Наоборот, они должны рассматриваться как важнейшее понятие медицины и самостоятельный объект научного исследования. При этом следует учитывать, что теория медицинских алгоритмов пока еще не существует, ее предстоит создать (см. главу 77).

## ВЫВОДЫ

1. Необходимо различать два понятия: алгоритм (математический алгоритм) и алгоритмическое предписание. Последнее имеет внешнюю форму алгоритма, но содержит не до конца определенные шаги.
2. Строго говоря, медицинский алгоритм — это не алгоритм, а всего лишь алгоритмическое предписание.
3. Тем не менее, термин «медицинский алгоритм» необходимо сохранить в связи с исключительной важностью этого понятия.
4. Недостаток существующих медицинских алгоритмов — отсутствие необходимой точности, приблизительность, отсутствие важных подробностей и условий. Другой недостаток — отсутствие свойства определенности (однозначности), которое является обязательным для математических алгоритмов.
5. Указанные недостатки являются одной из причин врачебных ошибок.
6. Задача состоит в том, чтобы повысить качество медицинских алгоритмов и по возможности приблизить их свойства к идеалу — к свойствам математических алгоритмов.
7. Математический алгоритм содержит в себе все указания, необходимые для его выполнения. В отличие от него, медицинский алгоритм содержит лишь часть указаний, а остальные опускает, рассчитывая на знания и сообразительность профессиональных врачей.
8. Проблема неопределенности в медицинской литературе является актуальной, поскольку она неблагоприятно отражается на здоровье населения.
9. Понятие «медицинский алгоритм высокой точности» (дракон-алгоритм) вводится для того, чтобы кардинально повысить качество медицинских алгоритмов, уменьшить неопределенность, снизить вероятность врачебных ошибок и обеспечить безопасность пациентов.
10. Прежние (неточные, не полностью определенные) описания медицинских алгоритмов должны со временем исчезнуть и уступить место принципиально новому средству — *медицинским алгоритмам высокой точности*.

# Глава 4

## АЛГОРИТМЫ ГИППОКРАТА

### ВВЕДЕНИЕ

Наша цель — показать, что медицинский алгоритм выражает сущность лечебно-диагностического процесса и является одним из центральных понятий медицины.

Мы будем обосновывать и защищать эту мысль на протяжении всей книги.

Начнем с истории вопроса. История медицинских алгоритмов — это история медицинских действий и решений.

Переведем часы назад и попытаемся понять: когда на белом свете появились алгоритмы? Не исключено, что они зародились намного раньше, чем обычно принято считать.

Попутно решим еще одну задачу. Мы будем изучать тексты Гиппократа и иллюстрировать их на языке ДРАКОН. Сочинения Гиппократа удобно использовать для предварительного знакомства с этим языком. Все алгоритмы в данной главе изображены на ДРАКОНе.

### СУЩЕСТВОВАЛИ ЛИ ВО ВРЕМЕНА ГИППОКРАТА МЕДИЦИНСКИЕ АЛГОРИТМЫ?

Знаменитый греческий врач Гиппократ, которого считают «отцом медицины», жил две с половиной тысячи лет назад. Он родился на 460 лет раньше Христа на острове Кос в Эгейском море. Его имя связано с собранием медицинских трактатов, известным как Гиппократов кодекс<sup>6</sup> [87].

Достанем эту внушительную книгу и откроем раздел «О женских болезнях». На странице 624 читаем:

---

<sup>6</sup> Связка старинных рукописей, подобно Библии, создана не одним лицом, а рядом древних авторов. Согласно Новой философской энциклопедии, «Именем Гиппократа названа наиболее известная коллекция древнегреческих медицинских текстов — “Гиппократов сборник” (72 сочинения), собранная в Александрийском музее не позднее 3-го века до нашей эры, которая включает тексты различных школ» [187].

## ПОДЛИННЫЙ ТЕКСТ ГИППОКРАТА

Когда месячные задержаны, бывает боль внизу живота; больной кажется, что у нее там тяжесть; она жестоко страдает поясницей и боком. Если же месячные совершенно уничтожены вследствие какой-нибудь болезни или если они густы, вязки, клейки, сначала нужно очистить желудок верхом и низом, потом очистить матку при помощи пессария, извлекающего кровь; затем дают промежуток, после чего прописывают слабительное, действующее на кровь; пусть больная пьет также крепленое вино, настоянное на ветках сосны.

Если месячное истечение не происходит, ей может показаться, будто она беременна, и сношение с мужем болезненно, так что ей кажется нечто внутри, и в животе чувствуется тяжесть; живот выдается; у нее те же самые прихоти, как у беременной женщины [88].

Что представляет собой этот отрывок? Обычный медицинский текст? Или, может быть, древний алгоритм?

Разумеется, слово «алгоритм» Гиппократ не знает и не употребляет. Тем не менее, давайте подумаем: является ли описание в рамке алгоритмом?

Попытаемся сравнить текст Гиппократа с чертежом на рис. 5, где используется язык ДРАКОН. Легко заметить, что текст и рисунок имеют в точности одинаковый смысл. Это значит, что речь идет об одном и том же алгоритме. На рис. 5 алгоритм представлен в графической форме, а у Гиппократа тот же самый алгоритм записан в виде текста.

Если это верно, можно сделать важное замечание. Медицинские алгоритмы появились в глубокой древности — задолго до того, как человечество изобрело термин «алгоритм»<sup>7</sup>. Всесторонний анализ Гиппократова кодекса подтверждает это предположение.

## КАК ПЕРЕВЕСТИ АЛГОРИТМ ГИППОКРАТА В ГРАФИЧЕСКУЮ ФОРМУ

Любой хорошо описанный алгоритм можно преобразовать из текстовой формы в графическую. Для этого существуют принципы и правила.

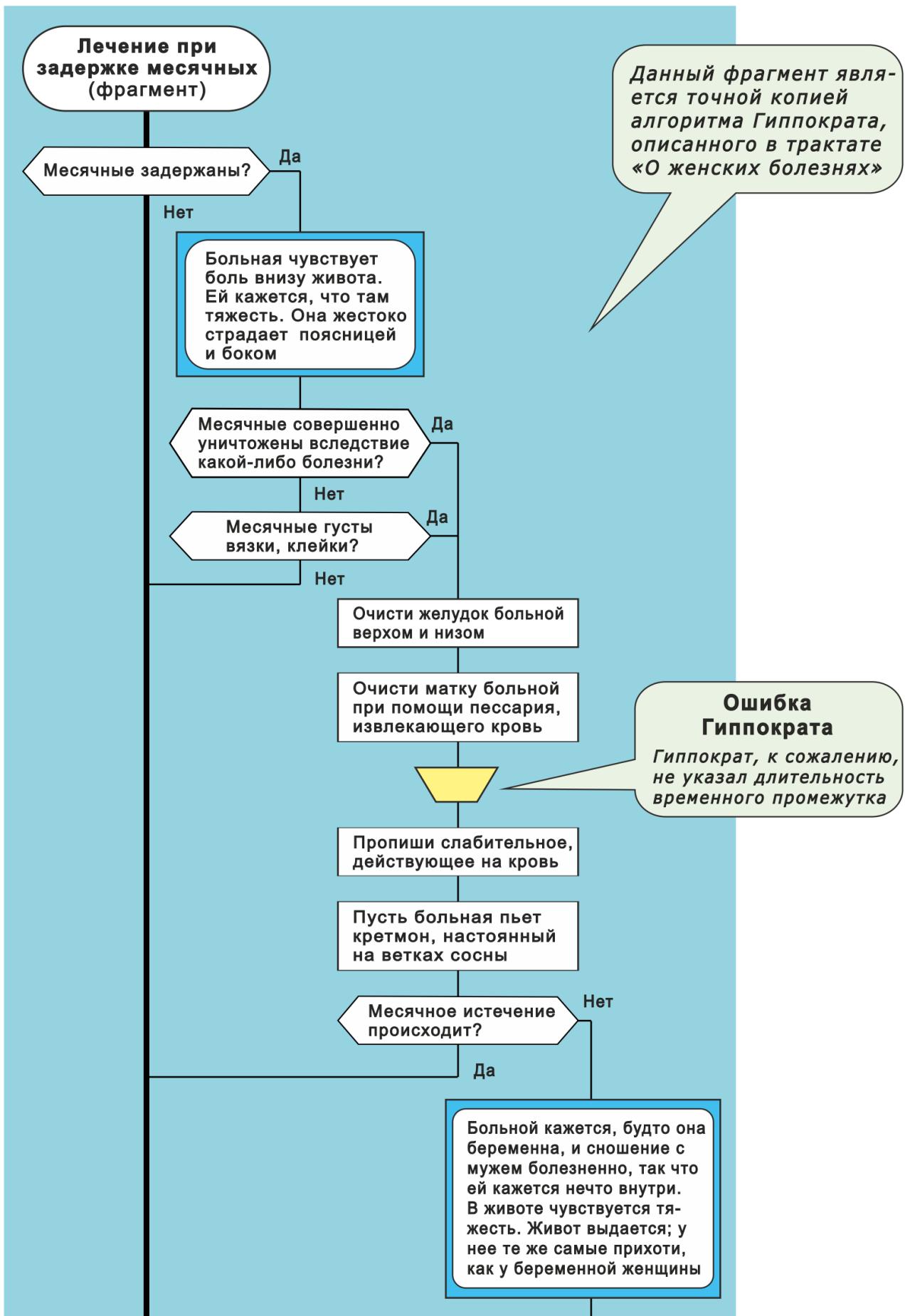
- Принцип 1. Выделите в тексте Гиппократа процедурные знания.
- Принцип 2. Разбейте выделенную часть на элементарные порции (шаги).
- Принцип 3. Каждый шаг поместите в отдельную графическую фигуру (икону).

Принципы определяют стратегию. Чтобы пользоваться ими на практике, нужны подробные правила.

Шаги алгоритма показаны на рис. 3 и 4. Шаги бывают разных типов. Гиппократ использует четыре типа:

1. Действия врача.
2. Условия (при которых выполняются действия).
3. Время (длительность паузы между действиями).
4. Комментарий к действиям (жалобы пациентки).

<sup>7</sup> Слово алгоритм образовано от имени персидского математика Аль-Хорезми (около 783 – 850).



**Рис. 5.** Медицинский алгоритм Гиппократа  
«Лечение при задержке menstrualных» на языке ДРАКОН [88].

Типы можно увидеть на рис. 5. Чтобы они были наглядными и сразу бросались в глаза, для разных типов применяют фигуры разной формы:

- действия пишут в прямоугольниках,
- условия — в шестиугольниках.
- время — в трапециях,
- комментарий — в фигурах с двойной рамкой.

## ПРАВИЛЬНЫЕ ТЕРМИНЫ

Для удобства читателя все рисунки в книге построены по единым правилам с помощью единой терминологии, присущей языку ДРАКОН. Познакомимся с терминами.

Фигуры на рисунках называются *иконами*. Внутри икон пишут текст. Пустые иконы запрещены.

В алгоритмах Гиппократа имеются четыре типа икон (см. выноски на рис. 6):

- икона *Действие*,
- икона *Вопрос*,
- икона *Комментарий*
- икона *Пауза*.

## ИКОНА «ВОПРОС»

Мы подошли вплотную к тому, чтобы раскрыть главный секрет Гиппократа. Осталось протянуть руку и достать заветную шкатулку. Но перед этим следует кое-что пояснить.

В верхней части рис. 7 изображена икона Вопрос. Она называется так потому, что внутри нее пишут «да-нетный вопрос». То есть вопрос, на который можно ответить либо «Да», либо «Нет». Все другие ответы запрещены.

Вот примеры да-нетных вопросов: рана кровоточит? Утюг сломался? Вася купил хлеб? Поезд пришел? Преступника арестовали? Тетя приехала? «Спартак» выиграл? Эта лужа больше, чем та? На улице температура выше нуля?

Икона Вопрос имеет один вход сверху и два выхода: вниз и вправо. Выход влево запрещен и никогда не используется (рис. 7).

Бывают случаи, когда нужно выбрать одно медицинское действие из двух. В этом случае удобно использовать икону Вопрос. При ответе «Да» выполняется действие «Зажми рану и подержи 5 минут». При ответе «Нет» — «Наложи повязку».

Икона Вопрос нужна, чтобы сделать в алгоритме разветвление. В этом легко убедиться, взглянув на рис. 7.

Иногда в алгоритме надо сделать не одну, а много разветвлений. Для этого используют несколько икон Вопрос — см. алгоритм Гиппократа на рис. 5.

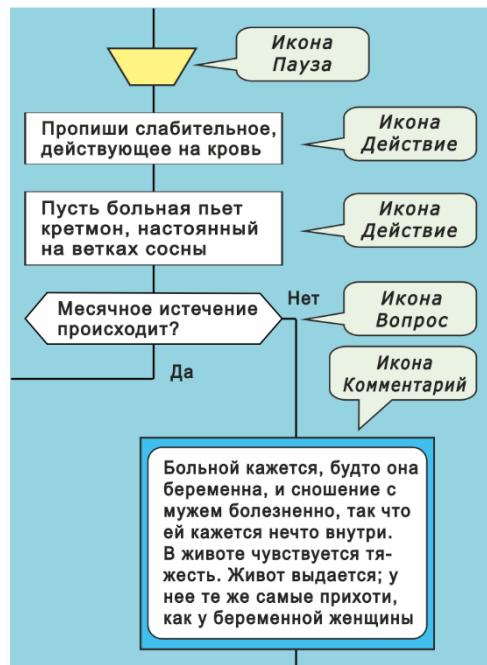


Рис. 6. Названия икон в алгоритме Гиппократа



Рис. 7. Икона Вопрос

## МЕДИЦИНСКОЕ РЕШЕНИЕ И ИКОНА «ВОПРОС»

Из главы 1 мы знаем, что медицинское решение можно изобразить с помощью иконы Вопрос (см. рис. 4, шаги 1, 2, 4, 7).

Медицинское решение имеет сложную структуру. Как показано на рис. 7, в его состав входят четыре элемента:

- икона Вопрос;
- текст вопроса, записанный внутри иконы;
- выход из иконы, помеченный словом Да;
- выход, отмеченный словом Нет.

Решение состоит в том, что врач должен выбрать один из двух выходов иконы Вопрос. Проще говоря, он решает: Да или Нет? Если у пострадавшего из раны льется кровь — Да. Если крови не видно — Нет.

Медицинское решение всегда принимают по отношению к конкретному больному.

## РЕШЕНИЕ И УСЛОВИЕ. ЧЕМ ОНИ ОТЛИЧАЮТСЯ?

Решение и условие — разные вещи. Парадокс в том, что в алгоритме они изображаются одинаково — с помощью иконы Вопрос. Как такое возможно?

Выше мы убедились, что да-нетный вопрос — один из четырех элементов решения. На рис. 6 таковым является вопрос: «Месячное истечение происходит?», на рис. 7 — вопрос «Рана кровоточит?»

*Любой да-нетный вопрос можно превратить в два условия.* Почему два? Потому что одно условие описывает выход Да, другое — выход Нет.

Вопросу «Рана кровоточит?» в алгоритме на рис. 7 соответствуют условия:

Условие 1

Если рана кровоточит

Условие 2

Если рана не кровоточит

Условие — это обстоятельство, от которого что-либо зависит. В данном случае от условия зависит медицинское действие.

*Ситуация* — это условие вместе с порождаемым действием. Алгоритм на рис. 7 описывает две ситуации.

Ситуация 1

Если рана кровоточит, зажми рану и подержи 5 минут

Ситуация 2

Если рана не кровоточит, наложи повязку

Сопоставим еще раз текст Гиппократа с алгоритмом на рис. 5. Нетрудно заметить, что медицинские решения (в тексте и графике) записываются по-разному. На рис. 5 решение показано в явном виде. В тексте, наоборот, оно представлено в скрытой, замаскированной форме — как условие в виде придаточного условного предложения. Важно подчеркнуть, что второе условие (которое обязательно должно учитываться) в тексте не указано, хотя и подразумевается.

Вооружившись этими знаниями, можно приступить к анализу фрагмента Гиппократа, записанного в рамке в начале главы.

Текст Гиппократа имеет четкую структуру. Вместе с тем, здесь есть подводный камень. Глядя на текст, человек как бы слепнет, потому что увидеть структуру глазами принципиально невозможно. Однако на рисунке она сразу видна во всей своей красе.

Ниже изложены правила перевода данного фрагмента в алгоритм. Правила оформлены в виде 13 шагов. Удобнее всего начать с выявления условий.

## КАК ПРЕОБРАЗОВАТЬ УСЛОВИЯ ГИППОКРАТА В АЛГОРИТМ

**Шаг 1.** Прочтите текст Гиппократа и выявите все случаи, когда встречаются слова «Когда» и «Если». Они указывают на УСЛОВИЯ.

**Шаг 2.** Выпишите у Гиппократа все условия. Легко убедиться, что в цитируемом отрывке Гиппократ пишет «Когда» и «Если» в четырех местах:

1. Когда месячные задержаны.
2. Если же месячные совершенно уничтожены вследствие какой-нибудь болезни.
3. Если они густы, вязки, клейки.
4. Если месячное истечение не происходит.

**Шаг 3.** Превратите условия в да-нетные вопросы, удалив слова «Когда» и «Если». Получим:

1. Месячные задержаны?
2. Месячные совершенно уничтожены вследствие какой-нибудь болезни?
3. **Они** густы, вязки, клейки?
4. Месячное истечение **не** происходит?

**Шаг 4.** Отредактируйте вопросы. Для этого замените местоимение «они» на слово «Месячные» и уберите отрицание «не»:

3. Месячные густы, вязки, клейки?
4. Месячное истечение происходит?

**Шаг 5.** Поместите каждый вопрос в икону Вопрос, как показано на рис. 5.

**Шаг 6.** Проставьте частицы Да и Нет в нужных местах.

Обратите внимание: у Гиппократа слова Да и Нет отсутствуют. Это значит, что в тексте Гиппократа явно описаны условия (но не решения). За шесть шагов мы превратили условия в решения.

Отсюда следует вывод: условия всегда можно превратить в решения.

## КАК ПРЕОБРАЗОВАТЬ ДЕЙСТВИЯ ГИППОКРАТА В АЛГОРИТМ

**Шаг 7.** Найдите у Гиппократа все ДЕЙСТВИЯ и выпишите их. В результате получим:

1. Сначала нужно очистить желудок верхом и низом.
2. Потом очистить матку при помощи пессария, извлекающего кровь.
3. После чего прописывают слабительное, действующее на кровь.
4. Пусть больная пьет также кретмона в вине, настоянном на ветках сосны.

**Шаг 8.** Превратите ДЕЙСТВИЯ в команды с помощью глагола в повелительном наклонении. Затем удалите лишние слова (словесный мусор): «сначала нужно», «потом», «после чего», «также».

1. Очисти желудок верхом и низом.
2. Очисти матку при помощи пессария, извлекающего кровь.
3. Пропиши слабительное, действующее на кровь.
4. Пусть больная пьет кретмон в вине, настоящем на ветках сосны.

**Шаг 9.** Поместите полученные команды в иконы Действие, как показано на рис. 5.

## КАК ПРЕОБРАЗОВАТЬ ВРЕМЯ У ГИППОКРАТА В АЛГОРИТМ

**Шаг 10.** Найдите у Гиппократа описания временных интервалов.

«Затем дают промежуток, после чего...»

**Шаг 11.** Поместите длительность временного промежутка в трапецию, как на рис. 5.

К сожалению, здесь Гиппократ допускает алгоритмическую ошибку. Величину времени он не указывает.

## КАК ПРЕОБРАЗОВАТЬ «ЖАЛОБЫ ПАЦИЕНТКИ» У ГИППОКРАТА В АЛГОРИТМ

**Шаг 12.** Найдите у Гиппократа описания жалоб больной. Получим:

1. Бывает боль внизу живота; больной кажется, что у нее там тяжесть; она жестоко страдает поясницей и боком.
2. Ей может показаться, будто она беременна, и сношение с мужем болезненно, так что ей кажется нечто внутри, и в животе чувствуется тяжесть; живот выдается; у нее те же самые прихоти, как у беременной женщины

**Шаг 13.** Поместите каждую жалобу в икону Комментарий, как показано на рис. 5.

Завершив выполнение всех шагов, мы получили точный графический образ алгоритма Гиппократа, представленный на рис. 5.

## РАЗВИЛКИ В АЛГОРИТМЕ ГИППОКРАТА

Для дальнейшего анализа нам понадобится часть алгоритма, извлеченная из рис. 5 и показанная на рис. 8.

Мы видим, что Гиппократ использует разветвления. Подавляющее большинство алгоритмов содержит их. Линейные (неразветвленные) алгоритмы (см. рис. 1), встречаются крайне редко.

Проблема в том, что разветвления трудны для понимания. Более того, они опасны и являются источником врачебных ошибок.



**Рис. 8.** Развилки в алгоритме Гиппократа

Во избежание неприятностей, нужно знать метод предотвращения ошибок. Начнем с того, что изучим правила изображения, чтения и понимания развилок.

## МАРШРУТЫ В АЛГОРИТМЕ ГИППОКРАТА

Алгоритм Гиппократа очень удобен для анализа. Попробуйте ответить на вопросы:

- Сколько маршрутов на рисунке 8?
- Как идет каждый маршрут?

Чтобы помочь читателю, подсказка дана на рис. 9. Мы видим три маршрута (три цветные стрелки). Каждый маршрут показывает движение сверху вниз — от начальной точки к конечной.

От чего зависят маршруты? Как они образуются?

Маршруты зависят от ответов на вопросы.

Если на оба вопроса дан ответ Нет, ситуацию описывает левый маршрут — лиловая стрелка, проходящая через точку А.

Если на верхний вопрос ответ Нет, а на нижний Да, возникает средний маршрут — зеленая стрелка через точку В.

Наконец, если на верхний вопрос ответ Да, получим правый маршрут — коричневую стрелку через точку С.

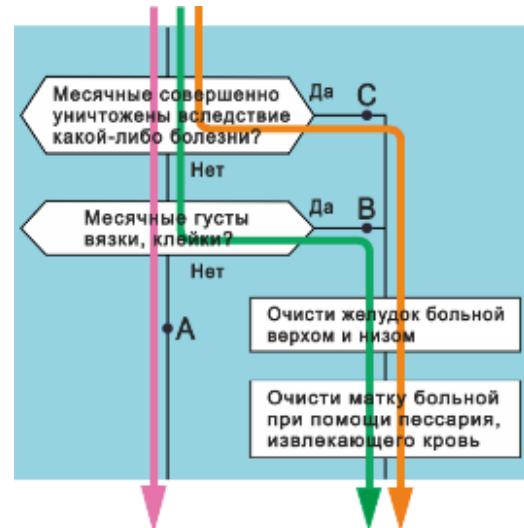


Рис. 9. Маршруты в алгоритме Гиппократа

## АНАЛИЗ КАЖДОГО МАРШРУТА

Маршруты следует анализировать по очереди. Сначала один, потом другой, затем третий. Чтобы не запутаться, изобразим каждый маршрут на отдельном рисунке.

Для этого выберем на рис. 9 левый (лиловый) маршрут и поместим его на рис. 10. Точно так же средний (зеленый) маршрут перенесем на рис. 11, а правый (коричневый) — на рис. 12.

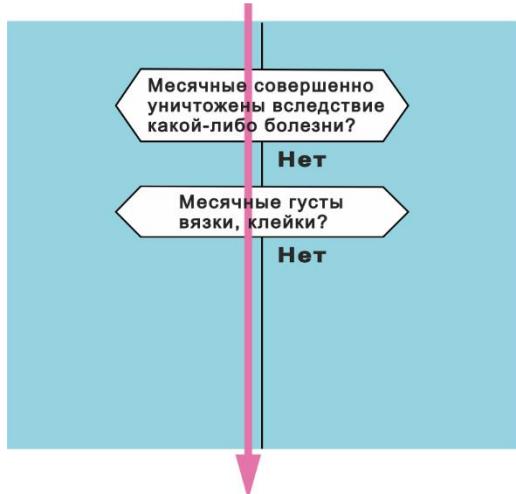


Рис. 10. Маршрут Нет–Нет



Рис. 11. Маршрут Нет–Да

Заодно мы осуществили «стриптиз» и удалили все лишнее. На рис. 10 у обеих икон Вопрос оставлен только один выход, а второй удален. Полученный маршрут можно назвать «Нет–Нет».

Это означает, что (на этом маршруте) лечащий врач уже принял два решения.

Верхнее решение «Нет» означает, что у больной месячные сохраняются.

Нижнее «Нет» указывает, что месячные нормальны; они вовсе не являются густыми, вязкими, клейкими. Это хороший, благоприятный для пациентки маршрут.

Остальные два маршрута, представленные на рис. 11 и 12, не благоприятны. Маршрут «Нет–Да» на рис. 11 говорит о том, что месячные, хотя и сохраняются, но явно ненормальны: они густы, вязки, клейки. Наконец, маршрут «Да» рис. 12 свидетельствует, что месячные «совершенно уничтожены» из-за болезни.

Как различать маршруты? По ответам на вопросы:

- Маршрут **Нет–Нет**.
- Маршрут **Нет–Да**.
- Маршрут **Да**.

Проведенный анализ хорош тем, что направлен на предотвращение врачебных ошибок. Врач должен назубок знать все маршруты алгоритма и понимать, чем они отличаются друг от друга. Если врач досконально изучит все эти тонкости, он вряд ли ошибется.

## ВОЗРАЖЕНИЯ МУДРОГО СКЕПТИКА

В этот момент дверь отворилась и раздался ироничный голос Скептика:

— Должен-то он должен. Врач много чего должен. А сможет ли он выполнить то, что должен? Является ли задача посильной для него?

— До сих пор, на протяжении всей истории медицины, — продолжал Скептик, — ученые заботились лишь о том, чтобы «запихнуть» в голову врача побольше знаний, упирая на ДОЛГ и подкрепляя его Клятвой Гиппократа. И никто не интересовался вопросом: А сможет ли врач выполнить свой ДОЛГ **без ошибок**?

На протяжении тысячелетий, включая почти весь XX век, статистика ошибок не существовала. Это значит, что обратная связь полностью отсутствовала.

— К чему это привело? — возвысил голос Скептик. — К тому, что врачебные ошибки превратились во всемирное бедствие. Благодаря докладам Института медицины США выяснилось, что медицина нередко убивает пациентов. Врачи ошибаются так же часто, как и все остальные люди. Попытка защититься от ошибок с помощью долга, клятв и наказаний недостаточна. Предложения Института медицины также не достаточны.

— Врачи живые люди, и они нуждаются в помощи. Помощь нужна, чтобы не ставить врачей в ложное положение. Чтобы помочь им сохранить чувство собственного достоинства. Врачи любят свою работу и гордятся ею. Медицинские ошибки, от которых пострадал пациент, являются сильнейшей психологической травмой для врача. Необходимо сделать все возможное, чтобы защитить врачей от подобных травм.

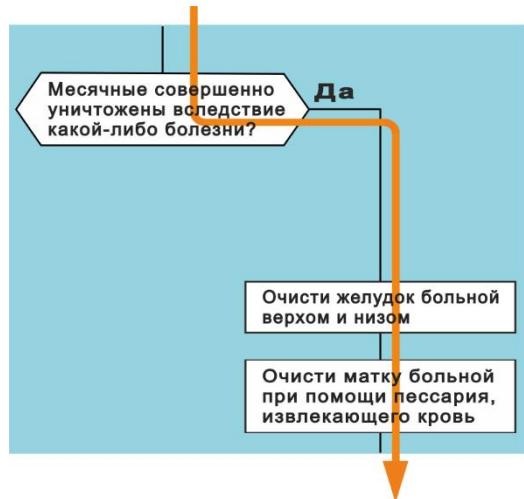


Рис. 12. Маршрут Да

Для этого прежние средства и методы не достаточны или не пригодны. Нужны дополнительные, специально разработанные меры, позволяющие уменьшить вероятность ошибок. Приблизиться к цели позволяет метод частных улучшений. Одним из таких средств является графическое представление алгоритмов. Можно показать, что графика — хорошее подспорье для защиты от врачебных ошибок.

## ПРЕИМУЩЕСТВА ГРАФИЧЕСКОГО АЛГОРИТМА

Мы провели тщательный анализ маршрутов с помощью пяти рисунков (рис. 8–12). Причем мы подвергли анализу лишь маленький кусочек алгоритма на рис. 5.

Вернемся к истокам и еще раз сравним текст Гиппократа (в рамке) и графический алгоритм на рис. 5. Можно ли проследить маршруты алгоритма, глядя на текст? Нет, это невозможно. Час смотри, два смотри, все равно ничего не увидишь.

Повторим: врач, читая текст, как бы слепнет, потому что текст скрывает маршруты и делает их невидимыми. Иная ситуация на рис. 5. Маршруты отчетливо видны. Их можно анализировать. По каждому маршруту можно провести пальцем, карандашом или указкой от начала до конца.

В электронном варианте можно обойтись без пальца. Глядя на экран компьютера, можно проследить как подсвеченная икона, словно солнечный зайчик, перемещается по заданному маршруту от начальной точки к концу алгоритма. Электронным способом можно без труда выявить и в динамике показать все маршруты.

Чем хорош графический алгоритм по сравнению с текстом? Он позволяет быстрее понять и легче запомнить материал. По этой причине его следует предпочесть в качестве средства предотвращения врачебных ошибок.

Медицинские алгоритмы должны войти в плоть и кровь каждого врача

## СЛОЖНЫЕ УСЛОВИЯ У ГИППОКРАТА

В медицинской литературе встречаются простые и сложные условия. Чем сложнее условие, тем больше вероятность путаницы и врачебной ошибки. Желательно расчленять сложные условия, чтобы превратить их в простые.

Рассмотрим составное условие в тексте Гиппократа:

«Если же месячные совершенно уничтожены вследствие какой-нибудь болезни **ИЛИ** если они густы, вязки, клейки...».

Нетрудно сообразить, что данная фраза состоит из двух простых условий:

1. «Если же месячные совершенно уничтожены вследствие какой-нибудь болезни»
2. «Если они густы, вязки, клейки».

Условия соединены с помощью союза «или», который в данном случае обозначает логическую операцию **ИЛИ**.

Операция **ИЛИ** — самый простой случай логического умозаключения. Рассмотрим его подробнее.

## ПРИНЦИП «РАЗРЕЖЬ ВЕЛИКАНА»

Можно ли избавиться от сложного условия? Можно. Для этого служит принцип «Разрежь великана», где под великаником подразумевается сложное условие. Данный принцип позволяет уменьшить сложность и снизить вероятность врачебной ошибки за счет:

- дробления текста на мелкие порции,
- преобразования текста в графику.

Выше мы расчленели сложное условие Гиппократа на две простые части.

Поедем дальше, превратим текст в графику и взглянем на рис. 5 и 8. Ни там, ни там мы не обнаружим сложного условия. Оно бесследно исчезло. Куда же оно делось?

Оно разбралось на две симпатичные дольки и превратилось в два вопроса. Причем каждый вопрос записан в отдельной иконе Вопрос.

В наглядной форме ситуация показана на рис. 13. Вместо сложного условия мы видим два маршрута, изображенные зеленой и коричневой стрелкой. В верхней части рис. 13 маршруты разветвляются, затем (после точек В и С) они сливаются. И порождают одни и те же действия:

- «Очисти желудок больной верхом и низом».
- «Очисти матку больной при помощи пессария, извлекающего кровь».

Мы убедились, что принцип «Разрежь великана» позволяет решить поставленную задачу: расчленить сложное условие в тексте Гиппократа и «уничтожить» его.

Зачем это сделано? Зачем мы раздробили сложное условие?

Сложное условие неудобочитаемо и провоцирует ошибки. Читатель может понять смысл неправильно. Чтобы этого не случилось, в языке ДРАКОН предусмотрена возможность полностью исключить сложные условия.

## СЛОЖНОЕ УСЛОВИЕ В ИКОНЕ «ВОПРОС»

На рис. 14 нарочно показан плохой пример. Ошибка в том, что в иконе Вопрос записан сложный вопрос, эквивалентный сложному условию. Две части вопроса связаны союзом «или».

Что же здесь плохого? Сравним рисунки 8 и 14, которые имеют одинаковый смысл. На рис. 8 сложное условие удалено и заменено на два простых вопроса. На рис. 14, напротив, сложное условие оставлено в неприкосновенности; оно «втиснуто» в одну икону Вопрос.

В качестве критерия для сравнения выберем длину вопроса, то есть число слов в вопросительном предложении. На рис. 8 два вопроса длиной 6 и 4 слова соответственно. На рис. 14 один вопрос длиной 11 слов.

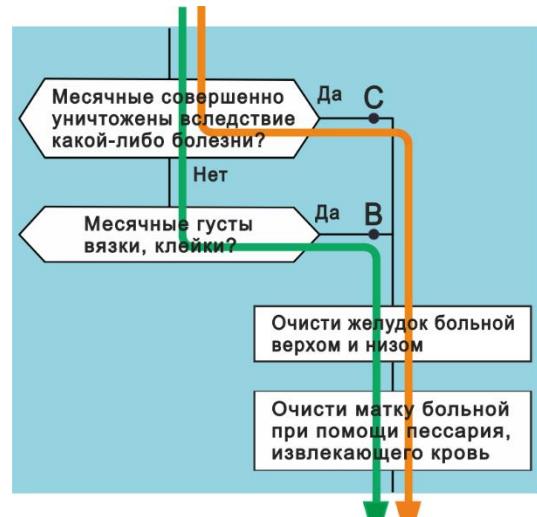


Рис. 13. Сложное условие Гиппократа превратилось в две иконы Вопрос

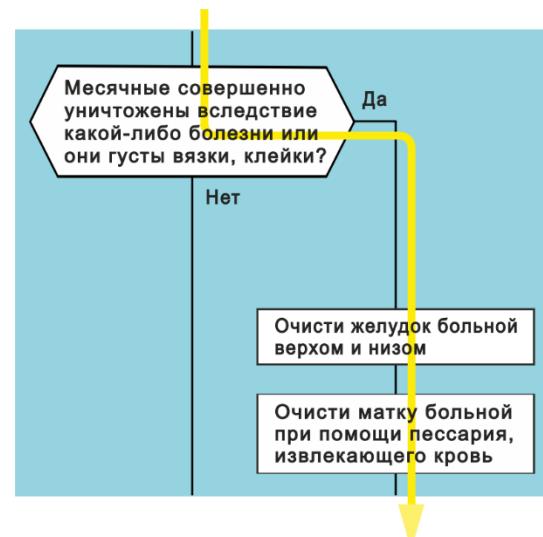


Рис. 14. Сложное условие Гиппократа размещено в одной иконе Вопрос

Правило 1

В иконе Вопрос желательно иметь минимальное число слов

В соответствии с Правилом 1 длинный вопрос (11 слов) на рис. 14 является нежелательным; следует предпочесть два коротких вопроса, показанные на рис. 8.

### Правило 2

- В иконе Вопрос нельзя писать сложное условие.
- Сложное условие надо раздробить на простые условия.
- Каждое простое условие следует преобразовать в вопрос и записать в отдельной иконе Вопрос.

В соответствии с Правилом 2 в языке ДРАКОН рекомендуется полностью исключить логическую операцию **ИЛИ** из текста, размещенного внутри иконы Вопрос. Больше того. Рекомендация касается не только операции **ИЛИ**, но и любых других, сколь угодно сложных логических операций.

В заключение сравним рис. 13 и 14, которые имеют одинаковый смысл. Два маршрута (зеленый и коричневый) на рис. 13 эквивалентны одному (желтому) маршруту на рис. 14.

Рисунок 13 является предпочтительным, потому что содержит простые вопросы, лишенные логических операций. И, наоборот, рисунок 14 следует считать нежелательным из-за чрезмерно сложного текста в иконе Вопрос.

## ОБЩЕИЗВЕСТНЫЕ СВЕДЕНИЯ ИЗ АЛГЕБРЫ ЛОГИКИ

(Пропустите этот параграф)

Базовыми элементами, которыми оперирует алгебра логики, являются высказывания.

Высказывания строятся над множеством  $\{B, \neg, \wedge, \vee, , 0, 1\}$ , где  $B$  — множество, состоящее из двух элементов:  $B = \{ \text{Истина}, \text{Ложь} \}$ .

Над элементами множества  $B$  определены три операции:

- $\neg$  отрицание (операция **НЕ**),
- $\wedge$  конъюнкция (операция **И**),
- $\vee$  дизъюнкция (операция **ИЛИ**),

а логический ноль **0** и логическая единица **1** — константы.

Как правило, в математических выражениях **Истина** отождествляется с логической единицей **1**, а **Ложь** — с логическим нулем **0**.

## В БЕЛОМ ПЛАЩЕ С КРОВАВЫМ ПОДБОЕМ... ПОНТИЙ ПИЛАТ И ИСТИНА

Истина сложное понятие. Понтий Пилат был первым, кто понял это. Беседуя с живым Богом, он поставил вопрос ребром: «Что есть истина?» [89].

Михаил Булгаков в «Мастере и Маргарите» не согласился с евангелистом Иоанном и придумал сказание об Иешуа Га-Ноцри, где Понтий Пилат обсуждает проблему истины на бытовом уровне:

— Зачем же ты, бродяга, на базаре смущал народ, рассказывая про истину, о которой ты не имеешь представления? Что такое истина? [90].

С тех пор прошло две тысячи лет, но на вопрос Пилата никто так и не ответил.

Математическая трактовка истины, принятая в алгебре логики, где истина объявляется логической единицей, слишком сложна и доступна далеко не всем.

Большинство людей, отвечая на вопрос «Деньги есть?», говорят «Да» или «Нет». И это правильно.

Однако программисты и другие специалисты, следуя правилам математической логики, поступают по-другому. Отвечая на тот же самый вопрос, они говорят: «Истина» или «Ложь». В данном случае термины «Истина», «Ложь» слишком трудны и запутывают дело.

В языке ДРАКОН этот недостаток устранен. И принято правило: логические понятия «Истина» и «Ложь» заменены на «Да» и «Нет» [91].

Слова да и нет намного проще и всем понятны. Они хорошо известны даже детям дошкольного возраста.

## ЛОГИЧЕСКАЯ ФОРМУЛА ГИППОКРАТА

Выделим еще раз небольшой фрагмент у Гиппократа и поместим его в рамку:

### Логическое рассуждение Гиппократа

Если же месячные совершенно уничтожены вследствие какой-нибудь болезни или если они густы, вязки, клейки, сначала нужно очистить желудок верхом и низом, потом очистить матку при помощи пессария, извлекающего кровь

Используя формализм Джона МакКарти [92], этот фрагмент можно записать в виде формулы (1):

**Логическая формула Гиппократа**

(1)

**Если (A V B = да) то С затем D**

где **A** — да-нетный вопрос «Месячные совершенно уничтожены вследствие какой-нибудь болезни?»

**B** — да-нетный вопрос «Месячные густы, вязки, клейки?»

**C** — действие «Очисти желудок верхом и низом».

**D** — действие «Очисти матку при помощи пессария, извлекающего кровь».

**V** — знак логической операции **ИЛИ** (дизъюнкция).

**то** — логическое следование.

**затем** — знак последовательности действий.

**Пояснение.** Формула **C** **затем** **D** означает следующее: сначала выполняется действие **C**, затем выполняется действие **D**.

Мы убедились, что рассуждение Гиппократа обладает «железной» логикой и может быть представлено в виде математической формулы.

Возникает вопрос: является ли формула (1) полезной для врачей?

На первый взгляд, нет, не является. Глядя на формулу (1), врач не получает никакой выгоды.

Однако первое впечатление обманчиво. Парадокс в том, что смотреть на формулу (1) нет необходимости. Она предназначена совсем не для того, чтобы на нее любоваться.

## НЕВИДИМАЯ МАТЕМАТИКА ТВОРИТ ЧУДЕСА

В языке ДРАКОН используется принцип «невидимой» математики, который позволяет (см. рис. 8 и 9):

- «тайком от врачей» использовать логическую формулу Гиппократа (1) в медицинском алгоритме Гиппократа с целью предотвращения врачебных ошибок;
- при этом формула (1) становится полностью невидимой;
- алгоритм Гиппократа работает в точном соответствии с формулой (1), несмотря на то, что увидеть эту формулу глазами невозможно;
- формула (1) убрана с глаз долой и «спрятана» в графических линиях алгоритма Гиппократа.

Это значит, что графика алгоритма Гиппократа на рис. 8 и 9 построена по строгим математическим законам в соответствии с логической формулой Гиппократа (1).

Мы объяснили *принцип невидимой математики* на конкретном примере, показанном на рис. 8 и 9. Следует отметить, что этот принцип обладает большими возможностями, о которых пойдет речь в последующих главах.

Преимущество в том, что при чтении ДРАКОН-алгоритма можно забыть об алгебре логики и ничего о ней не знать. Нужно лишь научиться проводить пальцем или карандашом от начала до конца алгоритма и при развилках уметь отвечать Да или Нет на простые (да-нетные) вопросы.

## СИСТЕМА МЕДИЦИНСКОГО ОБРАЗОВАНИЯ КАК ПЕРВОИСТОЧНИК ВРАЧЕБНЫХ ОШИБОК

Медицина древняя наука. Некоторые недостатки современной медицины берут начало в древности и частично сохранились до сих пор. Речь идет о вещах, порождающих врачебные ошибки и опасные для пациентов.

В основе современных представлений о борьбе с врачебными ошибками лежит неявное предположение, что причины ошибок кроются в медицинских учреждениях (*health care organizations*), а не в медицинском образовании.

На самом деле, это не совсем так. Ошибки действительно происходят в медицинских учреждениях, но глубинным источником многих (хотя и не всех) ошибок являются дефекты в системе медицинского образования. Имеются в виду дефекты медицинской литературы и медицинского языка.

В этой главе мы рассмотрим два недостатка и проиллюстрируем их примерами из Гиппократа:

- недостатки грамматико-стилистических средств,
- чрезмерная длина предложений.

## ГРАММАТИКО-СТИЛИСТИЧЕСКИЕ СРЕДСТВА

В современном учебнике по редактированию текстов, изданном под редакцией профессора Сусанны Антоновой, предназначенном для профессиональных редакторов, говорится:

«Большого внимания редактора требует оценка грамматико-стилистических средств изложения. Быстрому и адекватному восприятию информации обычно способствуют простые законченные предложения, имеющие правильную грамматическую форму. Сложные, громоздкие предложения с большим числом внутренних связей затрудняют понимание текста, вынуждая непроизводительно тратить время на его анализ и уяснение смысла. Поэтому сложные предложения, как правило, целесообразно расчленить на ряд простых, сохраняя логические взаимоотношения между ними» [93].

Профессиональные редакторы в медицинских издательствах предпринимают значительные и порою героические усилия, стремясь навести порядок в сложных

текстах, чтобы выполнить заветы Сусанны Антоновой. Но результаты, увы, очень скромные.

В медицинской литературе во многих случаях (а если честно, то сплошь и рядом) присутствуют указанные грехи, такие как «сложные, громоздкие предложения с большим числом внутренних связей». Это значит, что студенты-медики и врачи вынуждены, по словам Антоновой, «непроизводительно тратить время» на «анализ и уяснение смысла».

Поскольку время у студентов и врачей ограничено, чтение чрезмерно сложного текста приводит к тому, что смысл текста в ряде случаев оказывается не проясненным и усвоенным лишь частично, поверхностно, не в полной мере. Именно это обстоятельство (недостаточное и неполное усвоение знаний) создает предпосылки для будущих врачебных ошибок.

Легко заметить, что подлинный текст Гиппократа (в начале главы) имеет множество грамматико-стилистических недочетов. Гиппократа можно извинить, потому что он не читал учебник под редакцией профессора Сусанны Антоновой.

Хуже всего то, что за два с половиной тысячелетия не удалось устранить грамматико-стилистические погрешности медицинских текстов, которые до сих пор являются глубинным источником врачебных ошибок и представляют несомненную опасность для пациентов.

## ЧРЕЗМЕРНАЯ ДЛИНА ПРЕДЛОЖЕНИЙ У ГИППОКРАТА

Вернемся еще раз к тексту Гиппократа, который находится в центре нашего исследования:

«Если же месячные совершенно уничтожены вследствие какой-нибудь болезни или если они густы, вязки, клейки, сначала нужно очистить желудок верхом и низом, потом очистить матку при помощи пессария, извлекающего кровь; затем дают промежуток, после чего прописывают слабительное, действующее на кровь; пусть больная пьет также кретмон в вине, настоящем на ветках сосны» [88].

Предложение имеет сложную структуру и является неудобочитаемым, в нем 50 слов. Это слишком много и очень плохо. При всем уважении следует признать, что отец медицины Гиппократ подал своим последователям дурной пример в области грамматико-стилистических средств изложения медицинских знаний.

Язык ДРАКОН позволяет протянуть асклепиаду руку помощи и устраниТЬ дефект с помощью метода «Разделяй и властвуй!». В самом деле, нечитабельный текст Гиппократа можно разрезать на 7 мелких ломтиков, каждый из которых размещается в отдельной иконе, как показано на рис. 15.

Что же мы получили? Вместо сложного и длинного предложения из 50 слов — на рис. 15 представлен изящный набор, состоящий из коротких фраз длиной от 4 до 8 слов. Таким образом, благодаря ДРАКОНу длина предложений уменьшилась более чем в шесть раз!

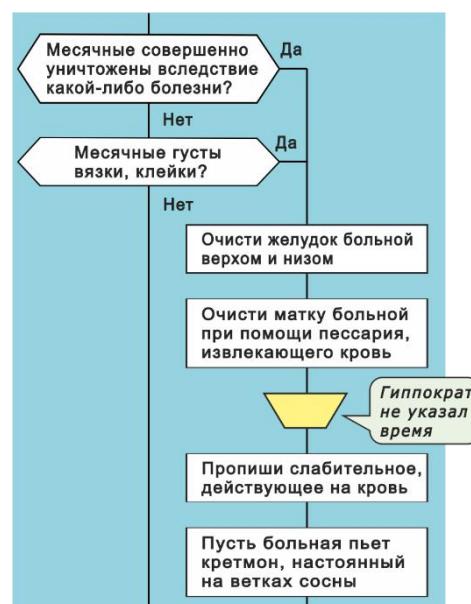


Рис. 15. Дробление сложного предложения Гиппократа на части

При этом *внутренние* связи исходного предложения, о которых обоснованно беспокоится профессор Сусанна Антонова, превратились во *внешние* связи между иконами. Тем самым невидимые связи, ранее недоступные для зрительного восприятия, обрели плоть и кровь — стали видимыми! Они преобразились в графические линии между геометрическими фигурами.

Следует подчеркнуть, что сплошной линейный текст превратился в *текст пространственный*. Это значит, что грамматические предложения графически разместились в разных местах пространства, причем точки размещения выбраны на основании правил когнитивной эргономики, что облегчает восприятие.

В итоге мы приобрели принципиально новое качество, так как произошла визуализация абстрактных понятий.

Данный пример убедительно показывает, что язык ДРАКОН приносит ощутимую пользу как средство для ПРАКТИЧЕСКОГО разукрупнения чрезмерно длинных процедурных предложений.

Но это не все. Попутно мы выявили еще одну оплошность. Фраза Гиппократа «затем дают промежуток, после чего» обозначает временной промежуток между двумя действиями. Чему же равен этот промежуток?

Об этом Гиппократ умалчивает, по-видимому, рассчитывая на знания или догадливость читателей. Ранее мы уже выяснили, что в алгоритмах подобные умолчания недопустимы. На рис. 15 ошибка отца медицины представлена в явном виде как пустая желтая трапеция, которую обязательно нужно заполнить конкретным значением длительности промежутка.

## ВРЕДНЫЕ РЕКОРДЫ

Мы убедились, что Гиппократ использует сверхдлинные трудные предложения, содержащие 50 слов. Дурной пример заразителен. В современной медицинской литературе можно найти сколько угодно неуместных подражаний отцу-основателю. Во многих книгах встречаются неудобочитаемые длинноты и иные препятствия, замедляющие процесс восприятия и осмыслиения. Вот пример:

«При восстановлении ритма сердца у больных с фибрилляцией предсердий при синдроме слабости синусового узла, синдроме преждевременного возбуждения желудочек тахисистолической формы для стабилизации гемодинамики и предупреждения периода асистолии показано использование временных однокамерных электрокардиостимуляторов с функциями учащающей стимуляции в режиме VVI и урежающей парной стимуляции в режиме ведения ритма, частой и сверхчастой стимуляции в режиме «burst» для купирования тахикардии (рис. 5.20, 5.21) [185].

В этом предложении 60 слов — на 10 слов больше, чем в грамматическом «рекорде» Гиппократа. Подобная негативная практика, как показывает тщательный анализ, широко распространена. Например, в учебнике «Поликлиническая терапия» ухитрились установить еще один рекорд и придумали предложение длиной в 74 (!) слова [94].

Такие или примерно такие неудобочитаемые сложные предложения являются характерной особенностью современной медицинской литературы. Все это создает питательную среду для врачебных ошибок и в конечном итоге отрицательно сказывается на здоровье населения.

## ДРАМАТИЧЕСКОЕ ПРОТИВОРЕЧИЕ МЕЖДУ МЕДИЦИНОЙ И ЛИНГВИСТИКОЙ

Мы сталкиваемся с удивительным парадоксом. С одной стороны, лингвисты, преподаватели и специалисты по редактированию текстов настоятельно рекомендуют избегать чрезмерно длинных предложений, потому что они создают неоправданные трудности для читателей, превращая тексты в кашу с гвоздями. Вспомним еще раз, чему учит профессор Сусанна Антонова:

«Сложные, громоздкие предложения с большим числом внутренних связей затрудняют понимание текста, вынуждая непроизводительно тратить время на его анализ и уяснение смысла. Поэтому сложные предложения, как правило, целесообразно расчленить на ряд простых, сохраняя логические взаимоотношения между ними» [93].

Антонову поддерживает главный редактор научного журнала профессор Сергей Чугров: «Слишком длинные предложения — враг внятности» и добавляет: «Пишите короче — будет яснее» [95].

Подобные рекомендации отнюдь не новы, они повторяются во многих работах, например:

«Следует избегать длинных, запутанных предложений. Трудна для восприятия мысль, выраженная одним предложением на целый абзац с массой словосочетаний и оборотов. Порой таких длинных и витиеватых, в конце которых автор иногда сам не может сообразить, что же он хотел сказать. Огрехи большей части подаваемых для публикации статей — слишком длинные предложения. Их лучше разбивать на отдельные фразы» [96].

«... не следует делать слишком длинные предложения, нагромождая их придаточными предложениями и деепричастными оборотами» [97].

«Понятность текстов можно улучшить различными способами. Первая рекомендация заключается в использовании коротких предложений. Так как в общем предложения тем понятнее, чем они короче» [98].

«Излагая материал, разделите его на понятные порции с помощью коротких предложений... Разбейте предложения, соединенные союзами или запятыми, таким образом, чтобы средняя длина предложений (average sentence length) была равна 15 слов или меньше»<sup>8</sup> [99].

С другой стороны, авторы важных публикаций, таких как медицинские учебники, стандарты, руководства, клинические рекомендации, протоколы, систематически нарушают разумные требования лингвистов и преподавателей. Можно сказать, что медицинские авторы игнорируют, не учитывают или «не слышат» настоятельные просьбы, требования и пламенные призывы ученых-филологов. Налицо отсутствие взаимопонимания между представителями двух авторитетных научных сообществ.

Приходится признать, что важнейшая междисциплинарная проблема, находящаяся на стыке медицины и лингвистики (проблема грамматико-стилистических средств изложения медицинских знаний) все еще недостаточно изучена и далека от разрешения.

Возникает вопрос: может быть, решить данную проблему принципиально невозможно? Нет, вполне возможно. Выше показано, что для случая процедурных

---

<sup>8</sup> Программа улучшения удобочитаемости текстов в науке и медицине PRISM (Program for Readability In Science and Medicine) [99].

медицинских знаний язык ДРАКОН позволяет эффективно расчленять сложные медицинские предложения и в разы сокращать их длину с помощью простого и детально описанного метода, результат которого представлен на рис. 15.

## ЧТО ТАКОЕ СЛИТНОЕ ПИСЬМО, или КАК ПИСАЛИ ДРЕВНИЕ ГРЕКИ

Современный читатель, раскрыв медицинскую или иную книгу, считает само собой разумеющимся, что текст аккуратно разбит на слова и предложения. Между словами повсюду расставлены пробелы. Границы предложений четко определены. Начало указано прописной буквой, в конце стоит точка или иной знак. В общем, повсюду полный порядок.

Но так было не всегда. Давайте проведем эксперимент и на минутку заглянем в прошлое.

СОВРЕМЕННЫЙ ЧИТАТЕЛЬ РАСКРЫВ МЕДИЦИНСКУЮ ИЛИ  
ИНУЮ КНИГУ СЧИТАЕТ САМО СОБОЙ РАЗУМЕЮЩИМСЯ ЧТО  
ТЕКСТ АККУРАТНО РАЗБИТ НА СЛОВА И ПРЕДЛОЖЕНИЯМ  
ЕЖДУ СЛОВАМИ ПО ВСЮ ДУРУССТАВЛЕНЫ ПРОБЕЛЫ ГРАН  
ИЦЫ ПРЕДЛОЖЕНИЙ ТОЖЕ ЧЕТКО ОПРЕДЕЛЕНЫ НАЧАЛО У  
КАЗАНО ПРОПИСНОЙ БУКВОЙ ВКОНЦЕ СТОИТ ТОЧКА ИЛИ  
ИНОЙ ЗНАК ВО ВСЕМ ПОВСЮДУ ПОЛНЫЙ ПОРЯДОК НО ТАК  
ЫЛО НЕ ВСЕГДА ВАЙТЕ ПРОВЕДЕМ ЭКСПЕРИМЕНТИНА  
МИНУТКУ ЗАГЛЯНЕМ В ПРОШЛОЕ

Попробуйте прочитать текст в рамке. Мы изобразили первые два абзаца в виде древнего маюскульного (унциального) письма. Читать, конечно, приходится медленно и с трудом. Но разобрать смысл можно.

Правила маюскула такие:

- все буквы заглавные, строчных нет,
- в каждой строке 40 букв,
- никаких пробелов между словами,
- никаких знаков переноса,
- перенос возможен в любом месте слова,
- никаких знаков препинания.

Две с половиной тысячи лет назад, во времена Гиппократа, в Греции было распространено именно такое *слитное* письмо. Оно называется «scriptio continua» [100]. Слова не отделялись друг от друга — они склеивались и слипались между собой, образуя непрерывную вереницу букв.

«Сочинения Гиппократа, вероятно, не дошли бы до потомства, если бы они не попали в Александрийскую библиотеку, основанную преемниками Александра Македонского, египетскими царями — Птоломеями» [101].

Было бы интересно посмотреть собственноручные тексты Гиппократа и оценить используемые им орфографические и грамматико-стилистические средства. К сожалению, это невозможно, так как оригиналы утрачены.

То, что дошло до нас — это результат труда многих поколений каллиграфов и монахов-переписчиков, берущий начало из экземпляров, некогда хранившихся в Александрийской библиотеке. Переписчики и редакторы вольно или невольно осуществляли модернизацию исходного текста, приводя его в соответствие с текущими грамматическими нормами. Так, например, «Гиппократов сборник» [53],

изданный в СССР в 1936 году под редакцией профессора Владимира Карпова, на который мы ссылаемся, соответствует нормам русского языка.

Поскольку подлинный текст Гиппократа отсутствует и показать его невозможно, в качестве примера *слитного* письма мы приводим греческий папирус Гомера, относящийся к 1-у веку до н. э. (рис. 16). Это маюскульное письмо, 30 букв в строке, без промежутков между словами и без знаков препинания.

Отсутствие знаков препинания объясняется просто — понятия «пунктуация» человечество еще не знало. После Гиппократа пройдет еще сотня лет, прежде чем греческий филолог Аристофан Византийский (257 – около 185 до н. э.) нащупает путь к созданию первых заслуженных пунктуации.

Выше было сказано, что у Гиппократа встречаются громоздкие предложения длиной 50 слов. Подобное утверждение справедливо для частного случая — русского перевода. Если же учесть, что утраченный оригинал Гиппократа был написан *слитным* письмом, то длина предложений могла быть намного больше.

Таким образом, грамматико-стилистические недостатки современной медицинской литературы, способствующие появлению врачебных ошибок, берут начало в глубокой древности и, несмотря на достигнутый прогресс, до сих пор остаются в значительной мере не исправленными и во многом неосознанными.

## ГОЛОС ДРЕВНЕЙ ИСТОРИИ

«Жизнь коротка, путь искусства долг, удобный случай скоропреходящ, опыт обманчив, суждение трудно», — это знаменитый афоризм Гиппократа, драгоценная жемчужина в сокровищнице мировой мудрости [102]. Начало его часто цитируется по латыни — «*Vita brevis, ars longa...*». Обращенный к потомкам завет Гиппократа до сих пор очаровывает сердца своим необыкновенным изяществом.

Отолоски этих слов с разными, порою красочными добавлениями и переливами встречаются и повторяются через многие сотни лет в книгах Луция Аннея Сенеки, в «Опытах» Монтеня и «Фаусте» Гете.

Монтень прочитал их глазами грустного скептика:

«Почти все древние философы утверждали, что нельзя ничего постигнуть, узнать, изучить, ибо чувства наши ограничены, разум слаб, а жизнь коротка» [103].

ΚΑΛΑΙΦΕΝΤΗΛΗΘΕΩΣΑΓΑΓΑΓΑΖ  
ΗΜΑΤΙΓΙΟΤΕΓΕΒΡΟΤΥΑΝΕΙΟΓΕΛΛΑΟΝΕ  
ΩΣΦΕΛΕΓΓΥΜΕΝΔΥΟΙΜΠΡΩΝΑΤΗΓΑΛ  
ΝΑΕΙΝΤΗΛΥΟΦΩΝΗΤΛΑΓΑΓΕΩΛΑ  
ΝΥΝΔΙΑΛΛΑΓΥΠΕΛΒΟΣΕΝΙΦΡΕΙΜΥΡΙΟΝ

Рис. 16. Греческий папирус со стихами Гомера выполнен *слитным* письмом без словоразделителей, что затрудняет чтение и анализ текста.  
1-е столетие до нашей эры [186]

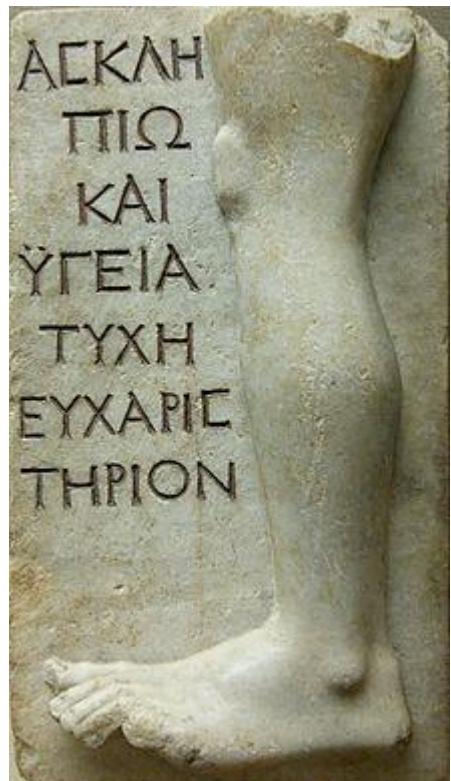


Рис. 17. Надпись в храме Асклепия (бога медицины и врачевания) на острове Милос в Эгейском море. Она содержит благодарность Асклепию за выздоровление после лечения больной ноги.

Пример маюскульного письма. Первоначально маюскул появился в виде изречений на камне и керамике и лишь затем на папирусе и пергаменте.

## ЯН КОМЕНСКИЙ ВОЗРАЖАЕТ ГИППОКРАТУ

К счастью, так думали не все. Нашелся человек, захотевший поспорить с Гиппократом, чтобы преодолеть «тупость нашего ума и неверность нашего суждения, которые служат причиной того, что мы обыкновенно занимаемся скорлупой, а в существо вещей не проникаем» [104].

Отец современного образования и автор «Великой дидактики» Ян Амос Коменский (1592 – 1670), которого Вильгельм Дильтей назвал «величайшим педагогическим умом, какой производила Европа», подверг серьезному анализу и критике афоризм Гиппократа [105].

Коменский заявил: «нашим стремлениям и нашей надежде может быть противопоставлено изречение Гиппократа: “Жизнь коротка, путь искусства долг, удобные случаи быстротечны, опыт обманчив, суждение о вещах трудно”... Гиппократ перечисляет здесь пять препятствий, служащих причиною того, что только немногие достигают высот науки» [106] [107].

Возражая Гиппократу, Ян Коменский посвятил значительную часть Великой дидактики обоснованию и развитию новых педагогических методов и средств, стремясь «открыть всеобщий, верный, легкий и прочный путь к изучению науки» [108] в том числе, для «развития умственных способностей, чтоб учиться было легко» [109].

## КРИТИКА МЕДИЦИНСКОГО ОБРАЗОВАНИЯ

Современная система преддипломного и последипломного медицинского образования отстала от жизни. Она во многом следует устаревшим воззрениям Гиппократа и далеко не всегда учитывает педагогические идеи Яна Коменского.

Для наших целей наиболее важна мысль Коменского об усовершенствовании языка. Он мечтал построить научно разработанный универсальный язык (*lingua catholica*). Позднейшие опыты Готфрида Лейбница по созданию универсального языка (*characteristic universalis*) «не только схожи с идеями Коменского, но в ряде случаев даже стимулированы Коменским, что подчеркивал сам Лейбниц» [110].

Мы связываем идеи Коменского о языке с его стремлением развить умственные способности людей, «чтобы учиться было легко» и с «развитием силы ума, дабы он легко проникал в существо вещей» [109].

Коменский впервые поставил задачу «быстро обучать всех всему» [111]. Он мечтал найти способ «легкого усвоения умом всех известных до сих пор искусств и наук» [111]. И требовал довести метод занятий до такого совершенства, чтобы при его помощи можно было овладеть знаниями с «наименьшим трудом» [111].

Подчеркнем эти слова — **овладеть знаниями с наименьшим трудом**. То есть, по выражению Рене Декарта, «без излишней траты умственных сил» [112].

Современное медицинское образование не удовлетворяет требованиям, которые сформулировал Ян Коменский, из-за серьезных дефектов профессионального медицинского языка.

Язык ДРАКОН позволяет в значительной мере устранить недостаток. Показано, что ДРАКОН улучшает работу ума [72].

## **МЕТАФОРА ЯНА КОМЕНСКОГО И МЕДИЦИНСКОЕ ОБРАЗОВАНИЕ**

Приведем высказывание Коменского об изучении наук. Слова, написанные 400 лет назад, можно во многом, хотя и не полностью, отнести к современному медицинскому образованию:

«Изучение наук является каким-то безысходным лабиринтом, в котором большинство бесцельно блуждает среди хаотического разнообразия вещей, среди какого-то океана, в котором легче утонуть, чем его исчерпать. Это делает изучение неотчетливым, так как изучаемое не приведено в достаточно прочный и очевидный порядок. Поэтому необходимо, собрав разбросанное, сжать его в малом объеме.

Средством против этого недостатка будет, если все, подлежащее изучению — крупное и мелкое — мы расположим в столь наглядном порядке, чтобы приступающие к изучению имели бы его перед собой так же отчетливо, как свои пальцы... Если они будут уверены в том, что весь океан образования они проплынут на том же самом корабле, на который они теперь всходят, вплоть до достижения желаемого совершенства» [113].

Язык ДРАКОН есть средство совершенствования процедурных медицинских знаний, позволяющий ускорить плавание через «океан образования». ДРАКОН дает возможность, «собрав разбросанное, сжать его в малом объеме», причем сделать это в «наглядном порядке, чтобы приступающие к изучению» видели суть дела ясно и отчетливо, как свои пять пальцев.

## **ДВА ВАЖНЫХ НЕДОСТАТКА**

Первый недостаток существующей системы медицинского образования состоит в том, что оно не имеет эффективных средств для предотвращения врачебных ошибок. По этой причине образование не может гарантировать безопасность пациентов. В этом отношении интерес представляют работы Института медицины США 2015 года в области диагностики [3] и межпрофессионального образования (interprofessional education) [114], которые, несомненно, являются полезными, но, к сожалению, далеко не исчерпывающими.

Второй недостаток — чрезмерная трудность, возникающая при изучении медицинской литературы (медицинских учебников, стандартов, руководств, клинических рекомендаций, протоколов). Это порождает «чрезмерные затраты времени» у студентов-медиков и врачей, «требует больших усилий от средних умов» [115].

Используя слова Коменского, можно сказать, что изучение медицинской литературы «должно происходить без преткновений и терниев, должно быть гладким, удобным», «без чрезмерных затрат времени», доступным для средних умов [115]. На самом же деле, подчеркивает Коменский, «уму приходится идти не последовательными ступенями, не по ровному пути, его ташат по кочкам, ямам, пещерам и расселинам» [116].

Гений Коменского прямо указывает на «тяготы и трудности учения», на то, что «школы превратились в места мучений», что система образования «устрашает своими хитросплетениями» и «предлагает человеку хлеб из камня, который ломает зубы и утомляет умы» [117].

По-видимому, пришло время еще раз обратиться к наследию великого педагога и осуществить серьезные изменения в системе медицинского образования,

направленные на повышение ее эффективности, чтобы студенты-медики и врачи могли овладеть знаниями с наименьшим трудом, без лишней траты умственных сил.

Язык ДРАКОН призван содействовать решению этой задачи и сократить чрезмерные затраты времени учащихся при одновременном повышении качества обучения.

## ВЫВОДЫ

1. Медицинский алгоритм выражает сущность лечебно-диагностического процесса и является одним из центральных понятий медицины.
2. Медицинские алгоритмы зародились в глубокой древности одновременно с возникновением медицины.
3. Анализ алгоритмов «Гиппократова сборника» показывает, что язык ДРАКОН правильно отражает алгоритмическую сущность процедурных медицинских знаний.
4. За истекшие два с половиной тысячелетия буквенно-цифровая и пунктуационная основа профессионального медицинского языка прошла большой путь развития от древнего рукописного маюскульного (унциального) письма до современной компьютерной письменности на основе Юникода [118] [119].
5. Недостаток в том, что грамматико-стилистические средства изложения медицинских знаний остаются неудовлетворительными, что препятствует быстрому и адекватному восприятию и осмыслению медицинской информации.
6. Серьезной и нерешенной проблемой медицинского образования является эргономичная визуализация абстрактных процедурных медицинских знаний, каковыми являются медицинские алгоритмы.
7. Медицинский алгоритмический язык ДРАКОН, в основе которого лежат педагогические идеи Яна Коменского об устранении неоправданных трудностей в учебе и овладении сложными знаниями с наименьшим трудом, предоставляет необходимые когнитивно-эргономические средства для системы медицинского образования.

Часть 2

# **МЕДИЦИНСКИЙ ЯЗЫК ДРАКОН**

## Глава 5

# ПРЫЖОК ДРАКОНА: ИЗ КОСМОСА В БОЛЬНИЧНЫЙ КОРИДОР

### КОСМИЧЕСКАЯ ОДИССЕЯ

В этой главе речь пойдет об истории создания медицинского языка ДРАКОН (рис. 18).

Судьба языка необычна. Он создавался вовсе не для медицины, а для космоса — в качестве языка программирования для орбитального корабля «Буран». При создании бортовых и наземных программ системы управления Бурана использовались языки Прол2, Диполь, Пси-Фортран, Лакс, Ассемблер (первые три разработаны в Институте прикладной математики имени М. В. Келдыша РАН). Обобщение опыта работы с этими языками привело к появлению ДРАКОНа [120] [121].

Цель разработчиков состояла в создании единого языка программирования и моделирования, который способен заменить специализированные языки: Прол2 (для разработки бортовых комплексных программ Бурана), Диполь (для создания наземных программ Бурана) и Лакс (для моделирования). Разработка языка ДРАКОН была завершена в 1996 году (спустя 3 года после неизбежного при крушении СССР, но грустного для участников закрытия программы Буран)<sup>9</sup>.

Жирограф и ДРАКОН Пилюгина

**Дружелюбный  
Русский  
Алгоритмический язык  
Который  
Обеспечивает  
Наглядность**

Космонавтика

Рис. 18. Кадр из документального фильма «Жирограф и ДРАКОН Пилюгина» [188].

<sup>9</sup> Автор участвовал в разработке Бурана с первого до последнего дня. В то время я был начальником лаборатории комплексной разработки вычислительной системы Бурана.

За последние двадцать лет с помощью ДРАКОНа (и основанной на нем технологии разработки алгоритмов и программ ГРАФИТ-ФЛОКС) были созданы системы управления многих крупных космических проектов: «Морской старт», «Протон-М», «Фрегат», «Наземный старт», «ДМ-03», «Ангара», южнокорейская ракета-носитель «KSLV» и др. [122] [123] [124].

Пуски ракет-носителей и космических разгонных блоков, при создании которых использовался и используется язык ДРАКОН, за истекший период выполнялись с пяти космодромов мира, расположенных на трех континентах (Европа, Азия, Америка) и в океане:

1. космодром Плесецк;
2. космодром Байконур;
3. Европейский космический центр во Французской Гвиане «Киру»;
4. южнокорейский космодром «Naro»;
5. международный плавучий космодром, производящий пуски с экватора в Тихом океане вблизи острова Рождества Республики Кирибати.



Рис. 19. Старт комплекса «Энергия — Буран» 15 ноября 1988 года с космодрома Байконур

## УДИВИТЕЛЬНОЕ И НЕОЖИДАННОЕ ПРОНИКНОВЕНИЕ В МЕДИЦИНУ

Альгирдас Карабюс (Литва) был первым, кто обратил внимание на возможность крупномасштабного использования языка ДРАКОН в медицине. Но не в качестве языка программирования, а совсем для других целей — в качестве графического средства для удобного описания последовательности действий врачей. То есть в качестве медицинского алгоритмического языка.

Инициативу Карабюса активно поддержали литовские врачи. За последнее время в Литве изданы четыре медицинских учебника на русском языке, в которых используется ДРАКОН [125].

1. Начальная неотложная акушерская помощь [126].
2. Специализированная реанимация новорожденного [127].
3. Неотложная медицинская помощь [128].
4. Травма [129].

Учебники апробированы в ряде стран (Литва, Казахстан, Азербайджан, Таджикистан, Туркменистан, Киргизия) в рамках курсов повышения квалификации врачей. Курсы проводили высококвалифицированные специалисты из Литовского университета медицинских наук (*Lietuvos sveikatos mokslų universitetas*) для местных врачей. Проведенная апробация дала положительные результаты [130].

## **ГУМАНИТАРНЫЕ ТРЕБОВАНИЯ К ЯЗЫКУ ДРАКОН**

Возникает вопрос: почему язык программирования ДРАКОН, созданный для ракетно-космической отрасли, оказался удобным средством для описания медицинских алгоритмов?

Это легко объяснить. При разработке языка была поставлена амбициозная цель: ДРАКОН должен не только удовлетворять практическим нуждам космических систем, но и решать широкий круг задач, выходящих далеко за рамки ракетно-космической техники и программирования [131].

В связи с этим при создании языка ДРАКОН были выдвинуты необычные для программистов и математиков гуманитарные требования:

1. Улучшить работу человеческого ума.
2. Предложить эффективные средства для описания не только алгоритмов, но и структуры человеческой деятельности в любой отрасли знаний (включая бизнес-процессы).
3. Предоставить человеку такие языковые средства, которые резко упрощают восприятие сложных процедурных проблем и общение с коллегами, делают непонятное понятным. И за счет этого буквально заставляют человека мыслить отчетливо, глубоко и продуктивно. В этих условиях вероятность заблуждений, просчетов и ошибок неизбежно падает, а производительность растет.
4. Радикально облегчить межотраслевое и междисциплинарное общение между представителями разных организаций, ведомств, отделов, лабораторий, научных школ и профессий.
5. УстраниТЬ или уменьшить барьеры взаимного непонимания между работниками различных специальностей (врачами и физиками, биологами и математиками, конструкторами и экономистами и т. д.), а также программистами и теми, кто не владеет программированием [132] [133].

## **ЯЗЫК ДРАКОН. МЕДИЦИНСКИЙ ВАРИАНТ**

Кроме того, была учтена медицинская специфика, чтобы профессиональные врачи могли легко и быстро описывать известные им, а также вновь создаваемые медицинские алгоритмы.

Для этой цели «космический» ДРАКОН был значительно сокращен и представлен в упрощенном виде, предназначенном специально для медиков и биологов. Были удалены все иконы (графические фигуры), связанные с программированием. И оставлены только те, которые нужны для медицины.

В результате появился биомедицинский вариант языка, так называемый «Медицинский ДРАКОН».

## **УДОБНЫЕ ГРАФИЧЕСКИЕ ИНСТРУКЦИИ ДЛЯ ВРАЧЕЙ**

Принято различать процедурные и декларативные знания (знания «как» и знания «что») [134]. Любые процедурные медицинские знания можно представить на медицинском языке ДРАКОН в виде графических инструкций для описания последовательности действий врачей. То есть в виде медицинских алгоритмов.

Примеры таких алгоритмов (инструкций) показаны в последующих главах.

## ЧТО ДУМАЕТ ВРАЧ О МЕДИЦИНСКОМ ДРАКОНЕ?

Травматолог-ортопед Василий Бачиашвили охарактеризовал медицинский ДРАКОН так:

По моему мнению, ДРАКОН отлично подходит везде в медицине, где нужно передать процедурные знания. Это описания биологических процессов, алгоритмы диагностики и лечения и т.д.

Было бы хорошо использовать ДРАКОН в методических пособиях и рекомендациях, в протоколах диагностики и лечения больных, в описании медицинских технологий, в медицинских книгах и журналах. Указанные материалы рассчитаны на то, чтобы большинство специалистов в данной области при чтении смогли их понять. Для того, чтобы в медицинской литературе начали использовать ДРАКОН, он должен стать максимально легким для освоения.

Облегченный вариант ДРАКОНа (медицинский ДРАКОН) мне очень нравится. Думаю, это движение в правильном направлении» [181].

## ВЫВОДЫ

1. Космическое происхождение медицинского языка сыграло важную роль, так как обеспечило тщательную отработку и надежность результатов, соответствующую высоким стандартам качества ракетно-космической отрасли.
2. В окончательном виде медицинский вариант языка ДРАКОН был сформирован в Литве по инициативе Альгирдаса Караплюса, литовских врачей и ученых, которые вложили огромный труд в его отработку и совершенствование.
3. Ученые и преподаватели Литовского университета медицинских наук (*Lietuvos sveikatos mokslų universitetas*) проанализировали «космический» ДРАКОН, упростили его, внесли полезные добавления и приспособили для медицинских нужд.
4. В Литве разработаны и опубликованы четыре медицинских учебника, в которых используется ДРАКОН.
  - Начальная неотложная акушерская помощь.
  - Специализированная реанимация новорожденного.
  - Неотложная медицинская помощь.
  - Травма.
5. Учебники апробированы в постсоветских странах: Литва, Казахстан, Азербайджан, Таджикистан, Туркменистан, Киргизия на курсах повышения квалификации врачей, которые проводили высококвалифицированные специалисты из Литовского университета медицинских наук.
6. Тщательные исследования языка ДРАКОН, проведенные литовскими врачами, и реальная практика его применения подтверждают целесообразность широкого использования языка для изображения медицинских алгоритмов в системе медицинского образования и в медицинской литературе (в медицинских учебниках, стандартах, руководствах, клинических рекомендациях, протоколах).

## Глава 6

# СПРАВОЧНИК: ГРАФИЧЕСКИЕ ФИГУРЫ ЯЗЫКА ДРАКОН

### ЗАЧЕМ НУЖЕН СПРАВОЧНИК

При анализе дракон-алгоритмов иногда возникает необходимость вспомнить, как называется та или иная фигура. И зачем она нужна.

Для этого служит справочник фигур, представленный в этой главе.

### ИКОНЫ МЕДИЦИНСКОГО ЯЗЫКА ДРАКОН

ДРАКОН — графический язык. Буквами этого языка являются геометрические фигуры, которые называются «иконы». Всего имеется 20 икон. Назначение икон показано на рис. 20.

Иконы должны быть заполнены текстом. Если текст отсутствует (икона пустая), значит, это ошибка.

Чаще всего используются две иконы: *Действие* и *Вопрос*.

В иконе *Действие* пишут команду в повелительном наклонении, например:

- Обеспечь проходимость дыхательных путей.
- Выполнни 30 компрессий грудной клетки.

В иконе *Вопрос* пишут да-нетный вопрос, т. е. вопрос, имеющий только два ответа: Да и Нет. Например:

- Есть ли реакция на прикосновение?
- Есть ли дефибриллятор?

	Икона	Название иконы	Пояснение
1		Заголовок	В иконе «Заголовок» пишут название медицинского алгоритма, например, «Измерение кровяного давления»
2		Конец	В этой иконе пишут слово «Конец»
3		Действие	Указывают действие, которое должен выполнить медицинский работник или прибор
4		Вопрос	Да-нетный вопрос, т. е. вопрос, на который можно ответить либо Да, либо Нет. Все другие ответы запрещены
5		Выбор	Фраза (или вопрос), приглашающая выбрать один из вариантов
6		Вариант	Здесь пишут один из вариантов. (Число рассматриваемых вариантов равно числу икон «Вариант»)
7		Имя ветки	Эта икона обозначает начало ветки. В ней находится название ветки. (Ветка — это структурная часть алгоритма)
8		Адрес	Икона «Адрес» обозначает конец любой ветки, кроме последней. Она показывает, в какую следующую ветку надо идти
9		Вставка	Икона «Вставка» говорит, что в этом месте из медицинского алгоритма вынут «кусок», который перенесён в другое место. В иконе пишут название этого «куска»
10		Пауза	Икона «Пауза» задерживает выполнение действия. Время задержки пишут внутри иконы
11		Время	В иконе «Время» пишут длительность выполнения действия (или решения)
12		Время группы	Длительность выполнения группы действий. Не одного действия, а именно группы. Группа состоит из двух и более действий
13		Время группы справа	Икона «Время группы» присоединяется справа
14		Начало контрольного срока	В иконе пишут контрольное время критической процедуры. Например, «30 сек».
15		Конец контрольного срока	Указывают окончание контрольного времени. Например, «Прошло 30 сек».
16		Комментарий	Комментарий — это не действие. Это различные пояснения и подсказки, которые помогают быстрее понять алгоритм
17		Начало совместной работы	Означает НАЧАЛО одновременных скоординированных действий двух врачей
18		Конец совместной работы	Означает КОНЕЦ одновременных скоординированных действий двух врачей
19		Пояснение	Любые сведения, поясняющие икону, находящуюся слева от данной
20		Соединитель	Икона «Соединитель» используется при переходе с листа на лист (когда медицинский алгоритм размещается на нескольких листах)

Рис. 20. Иконы медицинского языка ДРАКОН

	Макроикона	Название макроиконы	Пояснение
1		Развилка	Черные точки — это валентные точки. В эти точки можно вставлять иконы, например, икону «Действие». Хотя бы одна валентная точка должна быть заполнена.
2	 N = 2 N > 2	Переключатель (число вариантов 2 и больше)	Переключатель — это часть алгоритма, имеющая один вход вверху и один выход внизу. Внутри переключателя алгоритм разветвляется на несколько дорожек. Число дорожек равно двум и более.  Переключатель строится с помощью иконы «Выбор» и нескольких икон «Вариант».  Под каждой иконой «Вариант» имеется валентная точка.
3		Цикл	Цикл нужен для того, чтобы повторять действия. Повторение прекращается, когда будет выполнено условие. Условие записывают в иконе «Вопрос».
4		Веточный цикл	Веточный цикл нужен для того, чтобы повторять действия, расположенные в одной или нескольких ветках.
5		Действие с заданной длительностью	Справа нарисована икона «Действие». Слева к ней прицеплена икона «Время». Макроикона 5 показывает, что время действия жестко задано.
6		Решение с заданной длительностью	Справа нарисована «Развилка». Слева к ней прицеплена икона «Время». Макроикона 6 показывает, что врач должен принять решение за указанное время.
7		Длительность группы действий. Время слева	Справа нарисовано не одно действие, а группа, состоящая из двух (или более) действий. Слева к этой группе присоединена икона «Время группы». Макроикона 7 показывает, что время группы действий жестко задано.
8		Длительность группы действий. Время справа	То же самое, что в пункте 7. Отличие в том, что икона «Время группы» присоединена не слева, а справа.
9		Совместная работа врачей	Означает синхронную, скоординированную работы врачей, включая: — начало совместных действий, — выполнение совместных действий, — конец совместных действий.

Рис. 21. Макроиконы медицинского языка ДРАКОН

## МАКРОИКОНЫ МЕДИЦИНСКОГО ЯЗЫКА ДРАКОН

ДРАКОН имеет не только мелкие фигурки (иконы), но и крупные; они называются *макроиконами*.

Подобно тому, как слова состоят из букв, макроиконы (графические слова) слагаются из икон (графических букв). Медицинский ДРАКОН имеет 9 макроикон (рис. 21).

Иконы и макроиконы — это строительные блоки, из которых сооружаются медицинские дракон-алгоритмы.

## ВАЛЕНТНЫЕ ТОЧКИ

Важной частью макроикон служат *валентные точки* (на рис. 2 они показаны как черные кружки). В эти точки последовательно вводятся иконы и макроиконы, которые в совокупности образуют графический узор. После заполнения икон текстом узор превращается в дракон-алгоритм.

## МАРКЕРЫ МЕДИЦИНСКОГО ЯЗЫКА ДРАКОН

В сложном медицинском алгоритме могут появиться несколько веточных циклов. Как отличить их друг о друга?

Желательно иметь признак, позволяющий моментально, *по внешнему виду* обнаружить, где один цикл, а где другой. Примерно так же, как мы, не задумываясь, отличаем кошку от собаки и от вороны.

Отличительным признаком служит маркер. Предусмотрены четыре маркера, имеющие мнемонические названия: *сплошной*, *эллипс*, *вертикаль*, *горизонталь*.

Приятным сюрпризом служит тот факт, что маркеры расставляются в дракон-алгоритме не вручную, а автоматически. Этую операцию выполняет программа дракон-конструктор.

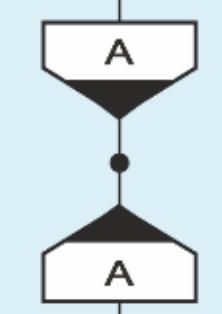
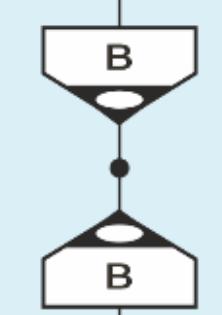
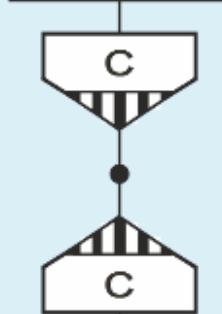
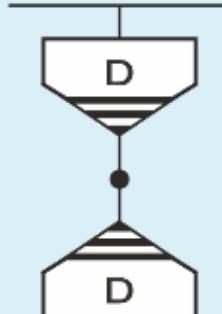
	Маркеры	Название маркера
1		Маркер 1 Сплошной
2		Маркер 2 Эллипс
3		Маркер 3 Вертикаль
4		Маркер 4 Горизонталь

Рис. 22. Маркеры нужны для того, чтобы легко различать веточные циклы при зрительном восприятии

## ДВА ЯЗЫКА

У медицинского ДРАКОНа есть старший брат, ракетно-космический ДРАКОН. Последний значительно богаче медицинского, у него вдвое больше икон и макроикон. Так что, если медикам понадобятся дополнительные выразительные средства, можно заимствовать нужные иконы у старшего брата.

## ВЫВОДЫ

1. Графические фигуры ДРАКОНа делятся на три части:
  - Иконы.
  - Макроиконы.
  - Маркеры.
2. Иконы и макроиконы — это мелкие блоки, из которых строятся дракон-алгоритмы.
3. Маркеры никак не влияют на правильность алгоритмов. Они облегчают чтение и зрительное восприятие наиболее сложных графических чертежей.
4. Данная глава представляет собой справочник. При анализе рисунков у читателя могут возникнуть вопросы: Как называется та или иная фигура. Глава позволит быстро разобраться и сэкономить драгоценное время.

## Глава 7

# ПРОСТЫЕ МЕДИЦИНСКИЕ АЛГОРИТМЫ. ПРАВИЛА И ПРИМЕРЫ

### ПРИМЕР МЕДИЦИНСКОГО АЛГОРИТМА

Проанализируем алгоритм на рис. 23. В верхней части в иконе Заголовок написано название алгоритма: «Реанимационные действия при наличии у новорожденного кистозной гигромы».

Чтобы прочитать алгоритм, надо проследить все тропинки, ведущие из Заголовка в Конец, и понять их смысл.

В данном алгоритме встречаются иконы: Действие, Вопрос, Вставка, Комментарий (см. подсказки на рис. 20).

Найдите три иконы Действие:

- «Применяй кислород в соответствии с показаниями пульсоксиметра».
- «Введи интубационную трубку глубоко, дальше сужения трахеи».
- «Транспортируй новорожденного в палату интенсивного наблюдения».

Прочтайте надписи в иконах Вопрос:

- «Есть кистозное образование в области шеи?»
- «Есть выраженная дыхательная недостаточность?»

Икона Комментарий украшена синей каемкой. Она содержит важное пояснение:

- «Нет клинических признаков кистозной гигромы.

### ИКОНА «ВСТАВКА»

Иногда бывает так, что на чертеже места уже нет, а работа все еще не закончена. Автор хочет любой ценой вставить в чертеж дополнительный кусок алгоритма. А места совсем нет. Как же быть?

Все очень просто. Не поместившийся кусок надо нарисовать в другом месте. А здесь, словно памятный знак, поместить икону Вставка. Она будет напоминать, что часть алгоритма «переехала» на новое место.

На рис. 23 икона Вставка снабжена надписью: «Использование пульсоксиметра». Это значит, что алгоритм с таким названием нарисован в другом месте, на рис. 24.

Слово «вставка» означает, что икону Вставка можно мысленно удалить, а вместо нее вставить нужный алгоритм.

Применительно к нашему случаю можно мысленно удалить икону «Использование пульсоксиметра» на рис. 23, а вместо нее вставить алгоритм, изображенный на рис. 24.

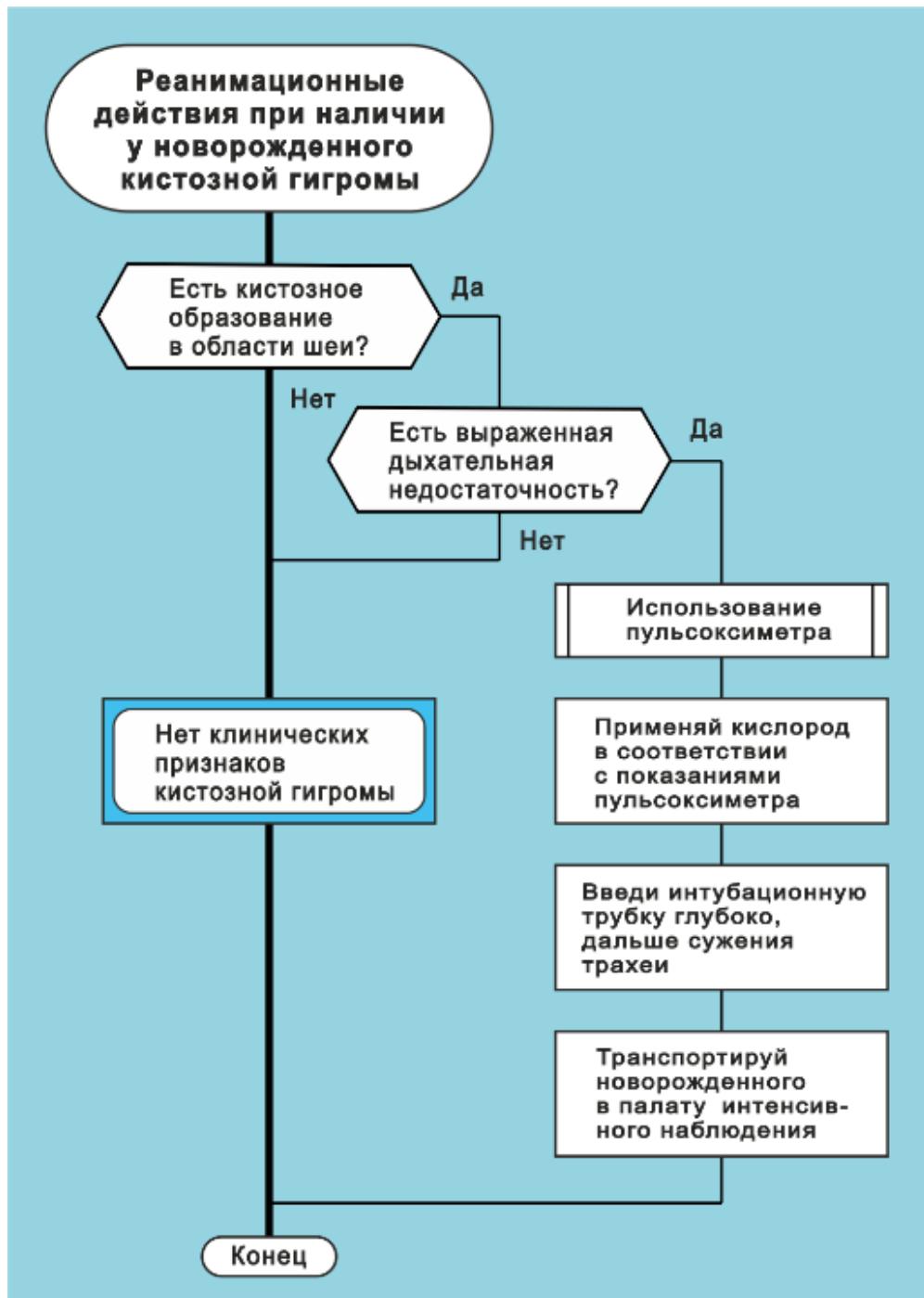


Рис. 23. Алгоритм «Реанимационные действия при наличии у новорожденного кистозной гигромы» [189].

Можно сказать по-другому. Вставка представляет собой икону-заместитель. Она замещает отсутствующий алгоритм и информирует, что на данном рисунке он не поместился. Поэтому пришлось «переселить» его на другой участок.

## ЧТО ТАКОЕ МАРШРУТ

Понятие «маршрут» нужно для того, чтобы создать в медицинском алгоритме образцовый порядок.

Дракон-алгоритм (рис. 23) имеет одно начало и один конец. *Маршрут* — путь, ведущий из начала в конечный пункт алгоритма. Все дракон-маршруты можно проследить пальцем от начала до конца, не отрывая палец от бумаги. Чертеж позволяет врачу одновременно видеть интересующие его маршруты.

Алгоритм на рис. 23 имеет три маршрута. (Найдите их!) Все они начинаются в иконе Заголовок и кончаются в иконе Конец.

Язык ДРАКОН предъявляет врачу в качестве подсказки полный обзор алгоритмической ситуации, т. е. полный набор маршрутов.

## ЧТО ЛУЧШЕ: ПОРЯДОК ИЛИ ПУТАНИЦА?

Чем хороша географическая карта? Тем, что она упорядочена. Вверху север, внизу юг. Смотрим вверх — видим север, смотрим вниз — видим юг. Двигаясь по меридиану сверху вниз, мы перемещаемся с севера на юг.

Важнейшая цель дракон-алгоритма — упорядочить алгоритмическую картину болезни и устраниТЬ путаницу. Для этого он выстроен по образцу географической карты.

Дракон-алгоритм на рис. 23 нарисован не как попало, а по строгим правилам. Вверху врач видит начало алгоритма и начало времени. Внизу — конец алгоритма и конец времени.

Двигаясь по алгоритму сверху вниз, мы перемещаемся во времени от начального момента до конечного.

Чтобы обеспечить в алгоритме дисциплину и порядок, введена специально разработанная эргономичная система понятий и правил [70].



Рис. 24. Алгоритм «Использование пульсоксиметра» [190].

Правило Дракона

- Любой дракон-алгоритм имеет только одно начало (один вход). И только один конец (один выход).
- Запрещено иметь в алгоритме несколько концов (несколько выходов).

Что такое маршрут

Это путь, идущий из начала алгоритма (из иконы Заголовок) до конца алгоритма (до иконы Конец)

## ВРЕМЯ ТЕЧЕТ СВЕРХУ ВНИЗ

Вертикаль играет важную роль в дракон-алгоритме. Время направлено вертикально вниз. На рис. 23 мы видим три вертикальных линии. На всех линиях время течет вниз.

Действия реализуются по правилу: «Чем ниже, тем позже». Действие «Транспортируй новорожденного...» выполняется после действия «Введи интубационную трубку...».

Благодаря использованию «вертикального времени» отпадает необходимость использовать стрелки.

Чтобы облегчить понимание алгоритма, вводится понятие бегунка. *Бегунок* — воображаемая точка, которая последовательно пробегает все иконы одного из маршрутов, перемещаясь из начала в конец.

Правило времени

Время в алгоритме течет сверху вниз (кроме циклов)

Порядок выполнения действий

- Действия на маршруте выполняются по принципу: «Чем ниже, тем позже».
- Действие, расположенное на маршруте ниже, выполняется позже, чем действие, расположенное выше

## ГЛАВНЫЙ МАРШРУТ И ШАМПУР

*Главный маршрут медицинского алгоритма* — наиболее желательный, наиболее благоприятный для больного путь от иконы Заголовок до иконы Конец.

Даже если алгоритм описывает печальные варианты, включая летальный исход, главный маршрут всегда описывает наиболее *благоприятный для пациента* вариант (из числа возможных).

На рис. 23 и 24 главный маршрут показан жирной линией.

*Шампур* — вертикальная прямая линия, соединяющая иконы Заголовок и Конец. Между ними на той же линии помещается одна или несколько других икон.

## ПРАВИЛО ГЛАВНОГО МАРШРУТА

Рассмотрим задачу. В запутанном лабиринте разветвленного медицинского алгоритма (рис. 23), нужно выделить один-единственный маршрут — царскую дорогу, путеводную нить. С ней можно зрительно сравнивать остальные маршруты, чтобы понять суть дела и не заблудиться в паутине развлечений и тропинок.

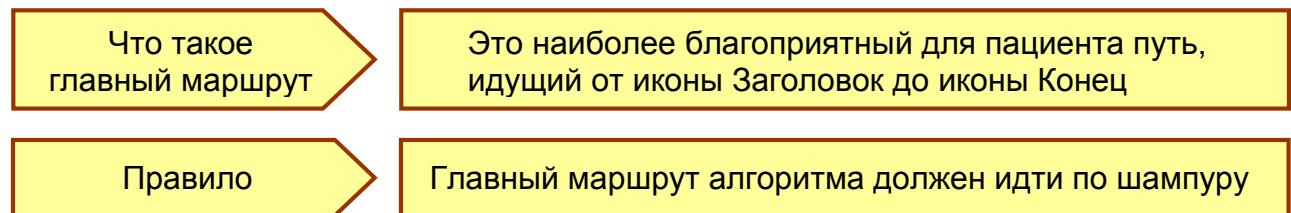
Путеводная нить (главный маршрут) должна быть легко различима, она должна сама бросаться в глаза. Бросив быстрый взгляд на дракон-алгоритм, врач сразу же видит царский маршрут и упорядоченные относительно него остальные маршруты.

Приятной новостью для медиков служит правило: «*Главный маршрут алгоритма должен идти по шампуру*».

Это значит, что царская дорога не может оказаться где-то на задворках медицинского алгоритма, где ее днем с огнем не сыскать. Нет, она всегда должна быть на самом почетном месте — на крайней левой вертикали.

Почему? Потому что крайняя левая вертикаль описывает самый хороший, самый предпочтительный для больного вариант развития заболевания или хирургической операции

Подобный порядок очень удобен. Он делает дракон-алгоритм четким, предсказуемым и интуитивно понятным.



## ИСПОРЧЕННЫЙ ГЛАВНЫЙ МАРШРУТ

Рассмотрим типичную ошибку. Предположим, правило главного маршрута не соблюдается (рис. 26).

Давайте проверим условие: «Рана испачкана?». Если рана чистая, это хорошо, если грязная — плохо. Главный маршрут всегда идет там, где хорошо. Следовательно, в развилке «Рана испачкана?», главный маршрут проходит через Нет, т. е. сворачивает с вертикали.

Отвечаем на второй вопрос: «Рана кровоточит?». Если крови нет, это хорошо, а если есть кровотечение — плохо. Главный маршрут любит, где хорошо. Поэтому, в развилке «Рана кровоточит?», он идет через Нет.

Получается, что на рис. 26 царский путь два раза отклоняется от шампур и начинает делать зигзаги. Это недопустимо.

На рис. 25 ошибка исправлена — главный маршрут идет по шампуру.

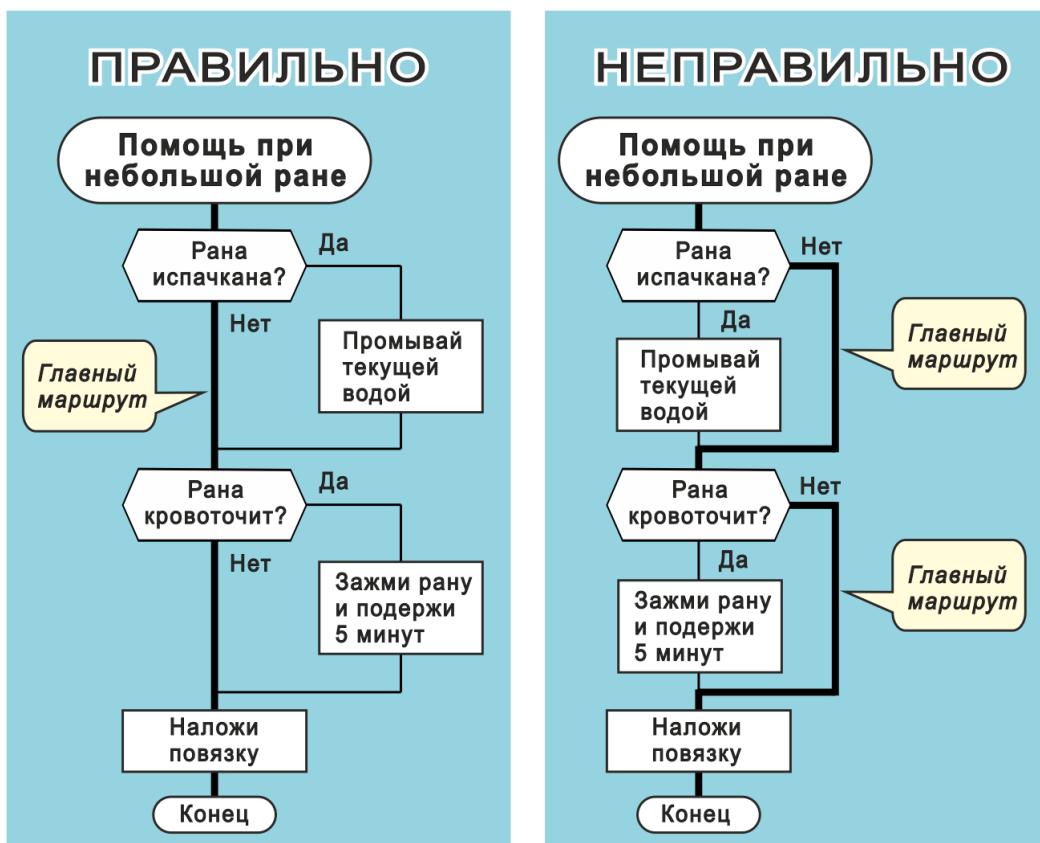


Рис. 25. Главный маршрут показан жирной линией. Правило главного маршрута выполняется. Поэтому царский путь прямой как стрела [191].

Рис. 26. Правило нарушено и главный маршрут «испортился», стал изломанным

## ВРАЧ ОБЯЗАН ЗНАТЬ ВСЕ МАРШРУТЫ АЛГОРИТМА

Рассмотрим простой вопрос. Сколько маршрутов на рисунке 25? Ответ: четыре. Все они показаны в таблице.

	Два ответа в верхней и нижней развилках	Как идет маршрут?
Маршрут 1 (главный)	Нет Нет	Маршрут идет по шампуру. Это главный маршрут, когда все благополучно (рана чистая и нет кровотечения).
Маршрут 2 (боковой)	Нет Да	В верхней развилке маршрут сворачивает направо через Да (рана грязная), а в нижней идет вниз через Нет (нет кровотечения).
Маршрут 3 (боковой)	Да Нет	В верхней развилке маршрут идет вниз через Нет (рана чистая), в нижней поворачивает направо через Да (рана кровоточит).
Маршрут 4 (боковой)	Да Да	В обеих развилках маршрут уходит вправо через Да. Это самый плохой вариант, когда рана грязная и кровоточит. Данный маршрут сдвинут вправо на обоих участках.

Таблица правильно отражает ситуацию, но у нее есть недостаток — она не наглядна.

Гораздо лучше и отчетливее видны маршруты на рис. 27. Рисунок показывает, что главный маршрут прямой как стрела. Остальные три маршрута боковые. Они расположены правее шампуря на одном или двух участках.

Обратите внимание: на рис. 27 все иконы Вопрос «одноногие»; у них показан только один выход. Это сделано не случайно.

## РАЗВЕРТКА АЛГОРИТМА

Каким образом получен рисунок 27? Чтобы выделить один маршрут из четырех на рис. 25, нужно в каждом разветвлении выбрать только один ответ, а второй отбросить и удалить.

Это значит, что мы *развернули* алгоритм на рис. 25, показали все дорожки по отдельности и расположили их по соседству (рис. 27). Полученный результат носит название *развертка алгоритма*. Она позволяет воочию увидеть все без исключения маршруты алгоритма.

Здесь есть одно но. Подобные развертки — слишком дорогое удовольствие. Чтобы построить развертку сложного алгоритма, нужно затратить много труда и времени. Это допустимо только как исключение и лишь для особо сложных случаев.

Во всех остальных случаях достаточно иметь обычный дракон-алгоритм (как на рис. 25) и уметь с ним работать, не прибегая к трудоемкой развертке.



Рис. 27. Развёртка алгоритма, представленного на рис. 25

На рис. 25 и 27 мы рассмотрели простейший разветвленный алгоритм. Тем не менее, тщательный анализ этого случая (при помощи развёртки) оказывается далеко не простым.

Проблема в том, что простые алгоритмы в медицине встречаются редко. Реальные алгоритмы, как правило, очень сложны.

Неумение анализировать сложные алгоритмы и распознавать их маршруты является причиной многих врачебных ошибок.

## АЛГОРИТМ УПОРЯДОЧЕН ПО ГОРИЗОНТАЛИ

Вспомним еще раз географическую карту. Движение взора по горизонтали упорядочено и имеет четкий смысл. Слева запад, справа восток. Смотрим налево — видим запад, смотрим направо — видим восток.

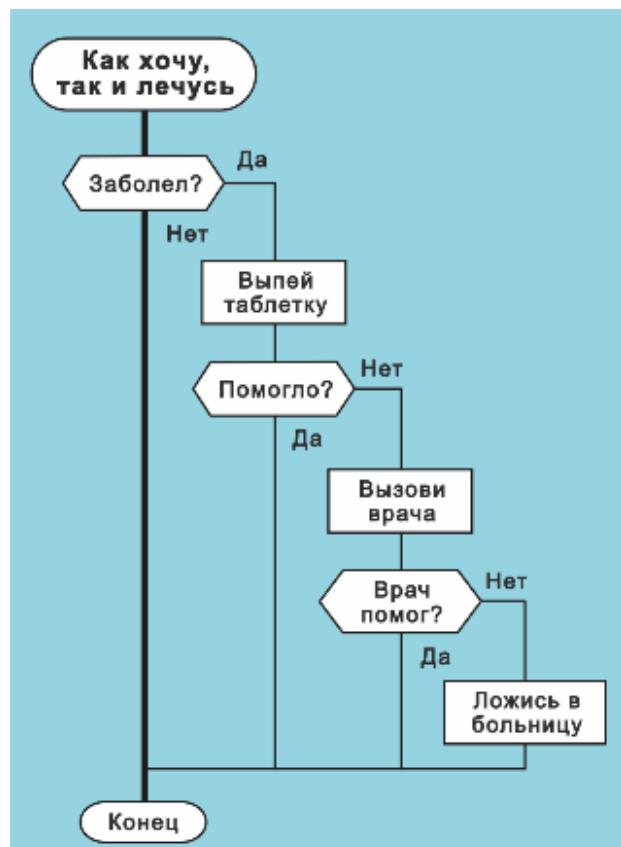


Рис. 28. Маршруты упорядочены слева направо

Дракон-алгоритм тоже упорядочен по горизонтали. Слева врач видит наиболее благоприятный для пациента маршрут, справа — наиболее тяжелый.

Двигаясь по алгоритму слева направо, мы перемещаемся от хорошей ситуации к плохой, от желательной — ко все более неприятной и угрожающей, вплоть до летального исхода.

Действует правило: «Слева хорошо, справа плохо», или «Слева лучше, справа хуже». Говоря более точно: «Слева более благополучный для больного исход, справа менее благоприятный».

## ПРАВИЛО БОКОВЫХ МАРШРУТОВ

Боковые маршруты нужно рисовать справа от шампера по принципу: «Чем правее, тем хуже».

Дракон-алгоритм превращает хаос в порядок. Для наглядности развернем эту мысль в шутливой форме на рис. 28. Все маршруты упорядочены согласно правилу: «Чем правее расположен маршрут, тем более неприятную ситуацию он описывает».

Левая вертикаль означает, что дела идут хорошо, ибо человек здоров. Вторая вертикаль описывает легкое недомогание, которое можно снять таблеткой. Третья вертикаль говорит: самочувствие ухудшилось, нужен врач. Наконец, крайняя правая вертикаль отражает самую неприятную ситуацию — пришлось лечь в больницу.

Что такое  
боковой маршрут

Это любой маршрут разветвленного алгоритма  
за исключением главного

Правило  
боковых маршрутов

Боковые маршруты алгоритма нужно рисовать справа от  
шампера по принципу: «Чем правее, тем хуже»

## КАРТОГРАФИЧЕСКИЙ ПРИНЦИП ЯЗЫКА ДРАКОН

Мы неоднократно подчеркивали, что дракон-алгоритм похож на географическую карту. Введем термин «Картографический принцип языка ДРАКОН». Принцип означает, что движение взора по горизонтали имеет строго определенный смысл. Слева находятся более благоприятные для пациента (хорошие) маршруты, справа — менее благоприятные (плохие).

Точно так же перемещение взгляда по вертикалам имеет четкий смысл: вверху начало времени, внизу — конец.

Картографический принцип — обобщающее и емкое понятие, которое включает в себя все понятия и правила, направленные на устранение визуальной путаницы и вводящие в дракон-алгоритм порядок и дисциплину. Сюда относятся: правило шампера, правило главного маршрута, правило боковых маршрутов, правило времени и т. д.

## ЧТО ТАКОЕ ПЕРЕКЛЮЧАТЕЛЬ

Схема на рис. 29 позволяет сделать в алгоритме разветвку на три направления. Для этого используются две иконы Вопрос. Однако задачу можно решить и по-другому — с помощью переключателя (рис. 30).

**Переключатель** — разветвление алгоритма на несколько тропинок, которые потом сливаются в одну. Переключатель сложная структура. Она строится из простых кирпичиков. Такими кирпичиками служат иконы Выбор и Вариант. Зачем они нужны?

Икона Выбор содержит вопрос или приказ, имеющий несколько ответов. Каждый ответ пишут в отдельной рамочке — иконе Вариант.

Взглянем на рис. 30. На первый взгляд там нет вопроса. Однако на самом деле вопрос есть, правда неявный. Чтобы убедиться, слово «светофор» прочитаем так: «Какой сигнал светофора сейчас горит?». Получим три ответа:

- зеленый,
- желтый,
- красный.

Попробуем описать, как работает переключатель. Описание обычно начинается со слова «если». Вот примеры.

- Если светофор зеленый — жми на газ.
- Если светофор желтый — притормози.
- Если светофор красный — стой (рис. 30).

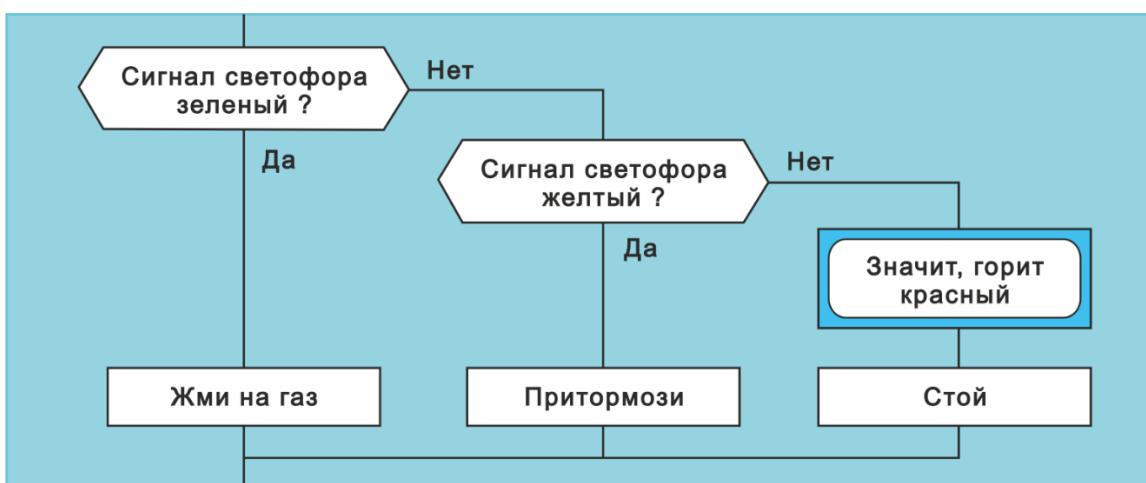


Рис. 29. Развилка на три направления, построенная с помощью иконы Вопрос

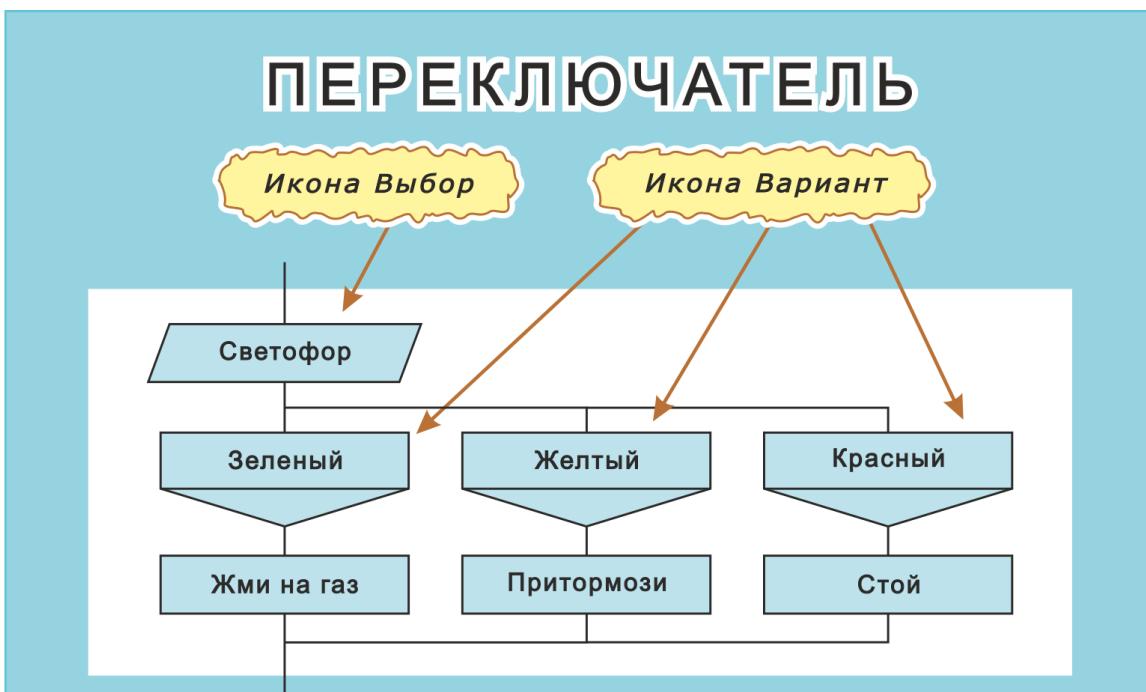
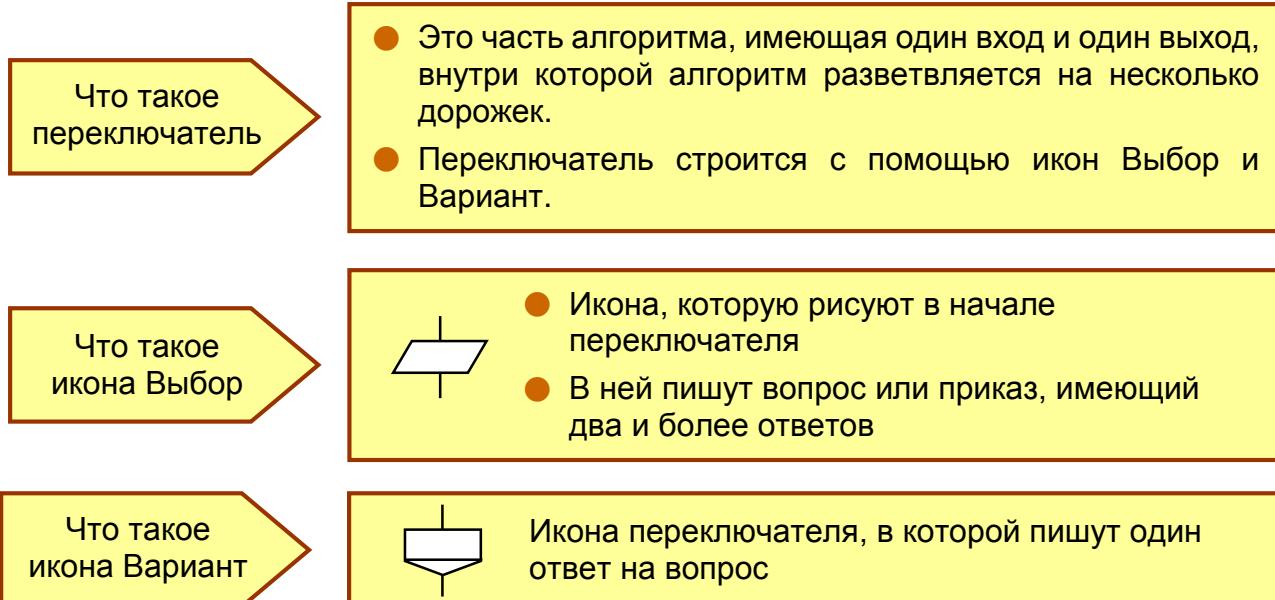


Рис. 30. Развилка на три направления, построенная с помощью переключателя



## ПЕРЕКЛЮЧАТЕЛЬ ДЛЯ ВЫБОРА МЕДИЦИНСКОГО ИНСТРУМЕНТА

В медицинских алгоритмах часто встречаются переключатели. Сначала рассмотрим уже знакомый нам пример. Он извлечен из рис. 2 и перенесен на рис. 31.

Вверху в иконе Выбор дано подробное указание (приказ): «Выбери средство для освобождения дыхательных путей». Иконы Вариант поясняют, что таких средств два: интубационная трубка и катетер. Стало быть, надо выбрать одно из двух: либо трубку, либо катетер.

При желании в иконе Выбор текст можно изменить и написать кратко: «Трубка или катетер?»

Внизу на рис. 31 в иконах Действие указаны конкретные команды: «Введи интубационную трубку...» и «Введи катетер для отсоса...».

Сколько шагов изображено в алгоритме на рис. 31? Два шага. Первый шаг — нужно принять решение, что будем использовать: трубку или катетер? Второй шаг — выполнить физическое действие.

В качестве упражнения давайте мысленно удалим из рис. 31 два действия и вместо них на вертикальных линиях поместим два черных кружка. Зачем? Об этом речь пойдет в следующем параграфе.

И последнее. Какой из маршрутов на рис. 31 более благоприятен для пациента: левый или правый? Ответить затруднительно. Следовательно, правило «Чем правее, тем хуже» здесь не применимо. Это значит, что мы столкнулись с исключением, но так бывает редко.

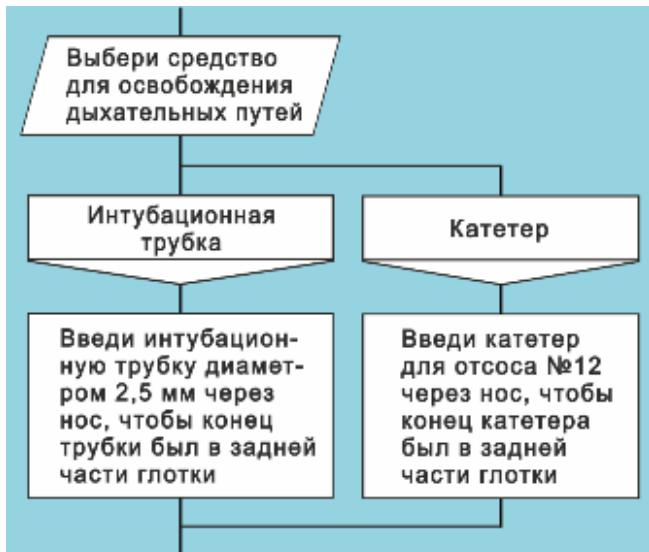


Рис. 31. Переключатель на два направления

В большинстве случаев принцип «Чем правее, тем хуже» эффективно работает и позволяет красиво расположить маршруты алгоритма. В этом можно убедиться при анализе примеров на рис. 32–34.

## ПЕРЕКЛЮЧАТЕЛЬ И ТЯЖЕСТЬ ЗАБОЛЕВАНИЯ

Переключатель позволяет указать число маршрутов и помогает врачу выбрать один из них. Для удобства читателя мы рассматриваем простейший случай, когда маршрутов всего два.

Выше мы изучили переключатель для выбора медицинского инструмента. На рис. 32 у переключателя иная функция — он учитывает тяжесть заболевания и отделяет легкий химический ожог от средних и тяжелых. В соответствии с этим маршруты четко упорядочены слева направо, причем более тяжелый ожог расположен правее. Это значит, что выполняется правило боковых маршрутов «Чем правее, тем хуже».

Обратите внимание: на рис. 32 показан только первый шаг (решение), а второй шаг (действие) скрыт и изображен условно, в виде черных кружков. Кружки обозначают валентные точки алгоритма. В эти точки чуть позднее мы введем иконы Действие или другие полезные вещи.

## ПЕРЕКЛЮЧАТЕЛЬ И ПОРАЖЕННЫЕ ОРГАНЫ

Незнакомый алгоритм гораздо легче воспринимается, если читатель заранее знает, что слева находятся «хорошие» маршруты, а справа «плохие», неблагоприятные для пациента.

Вот пример. Ожог конъюнктивы и роговицы глаза опаснее, чем ожог века. Поэтому эти два случая необходимо упорядочить слева направо по степени тяжести.

Правило боковых маршрутов «Чем правее, тем хуже» является хорошим критерием для этого случая. Оно позволяет внести ясность в алгоритм, избежать путаницы и облегчить изучение материала.

На рис. 33 в иконе Выбор записан вопрос: «Что повреждено?». В иконах вариант дан ответ: «Веки» и «Конъюнктива...»». Ниже в трех иконах Действие описаны соответствующие лечебные воздействия.

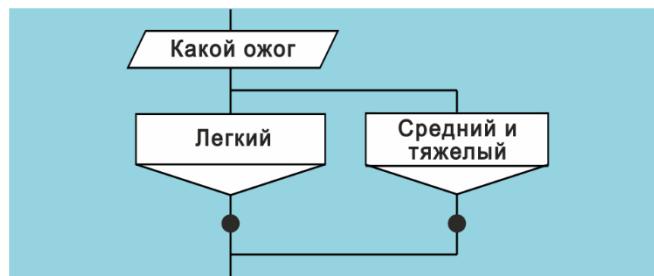


Рис. 32. Переключатель на два направления с валентными точками (черные кружки)

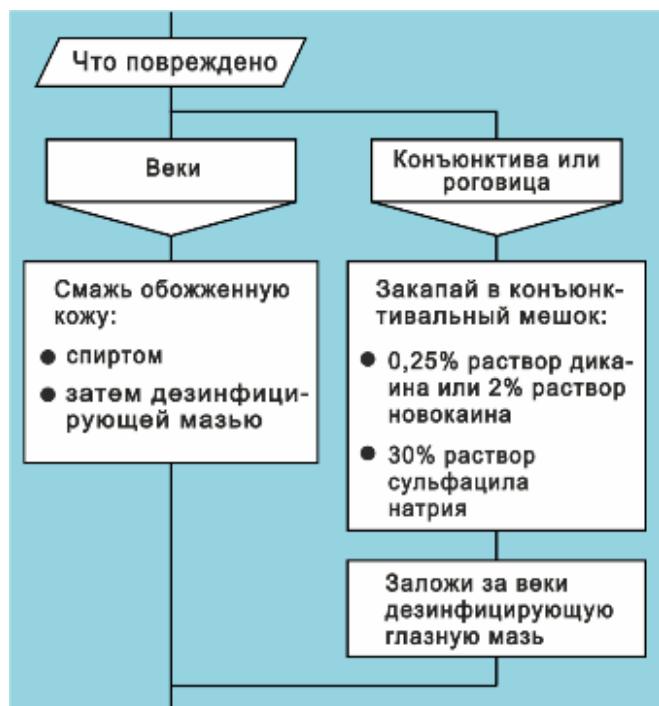


Рис. 33. Переключатель на два направления. Выполняется правило «Чем правее, тем хуже».

## ДВА ПЕРЕКЛЮЧАТЕЛЯ В ОДНОМ АЛГОРИТМЕ

Перейдем к более сложному случаю и рассмотрим алгоритм «Первая помощь при химическом ожоге глаз жидкостью». Кусочки этого алгоритма мы уже видели на рис. 32 и 33.

Подправим наш эскиз, объединим кусочки и добавим последний штрих. Результат показан на рис. 34. Чтобы понять структуру нового алгоритма, подскажем: он построен с помощью двух переключателей.

Построение выполняется за три этапа.

Этап 1. Выбираем переключатель на рис. 32.

Этап 2. В левый черный кружок (ниже варианта «Легкий») вставляем целиком переключатель на рис. 33.

Этап 3. В правый черный кружок вставляем три иконы Действие:

- «Введи подкожно...».
- «Введи внутримышечно...».
- «Наложи на поврежденный глаз...».

Выполнив эти операции, получим окончательный алгоритм на рис. 34. Нетрудно заметить, что он состоит из трех маршрутов. Главный маршрут, как и полагается, идет по шампуру. Он обозначен жирной линией.

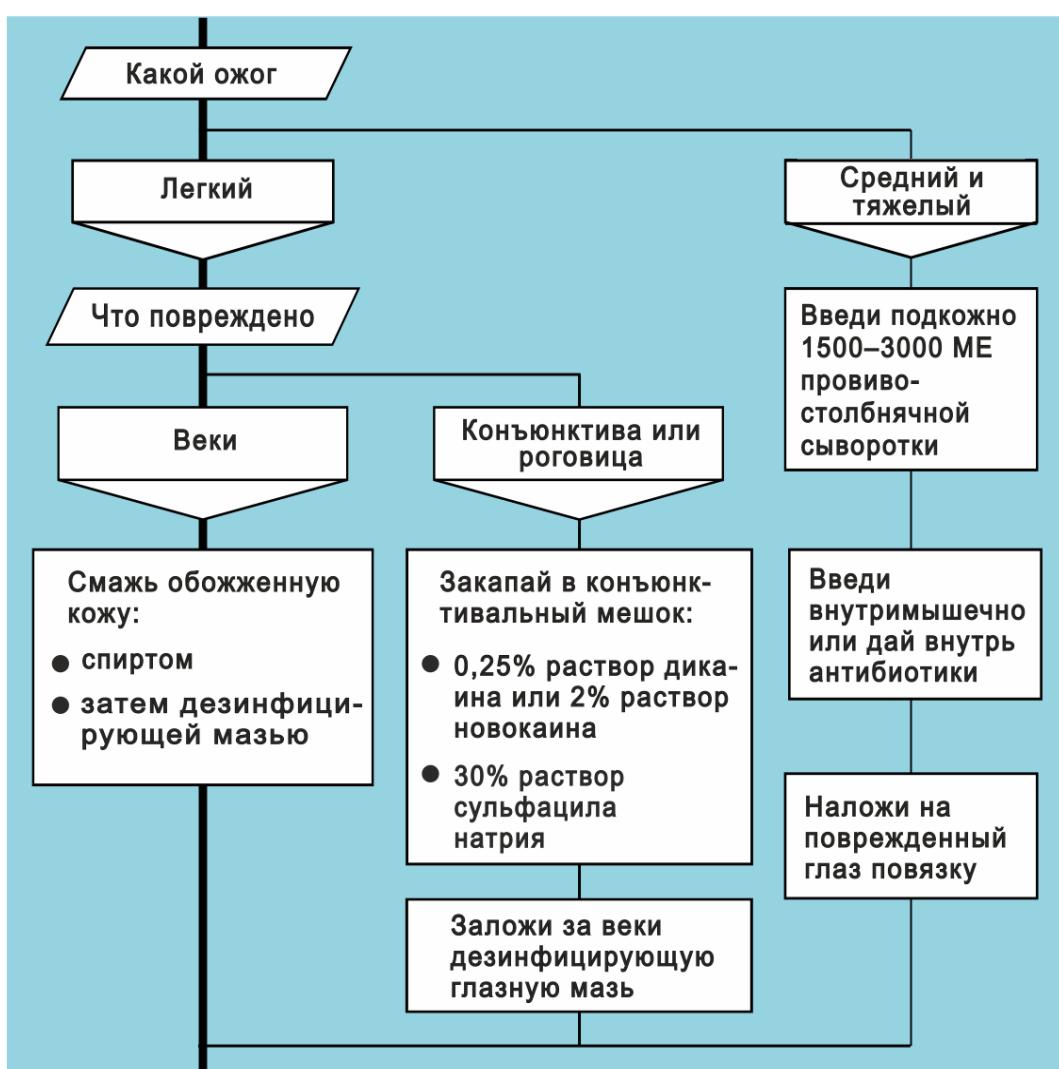


Рис. 34. Алгоритм с двумя переключателями

## МАКРОИКОНА ПЕРЕКЛЮЧАТЕЛЬ

Теперь пора вспомнить про макроикону «Переключатель». В медицинском языке ДРАКОН имеется девять макроикон, которые показаны в справочнике на рис. 21. Переключатель занимает там почетное второе место.

Итак, мы убедились, что на рис. 34 показан фрагмент первой помощи при химическом ожоге глаза с использованием макроиконы «Переключатель».

Верхний переключатель на рис. 34 содержит вопрос: «Какой ожог?» Две иконы Вариант предлагают выбрать один из двух ответов:

- «Легкий».
- «Средний и тяжелый».

Нижний переключатель спрашивает: «Что повреждено?» В иконах Вариант читаем ответы:

- «Веки».
- «Конъюнктива или роговица».

## ЧТО МЫ УЗНАЛИ В ЭТОЙ ГЛАВЕ

ДРАКОН — графический язык. Он имеет графические буквы. Необходимо запомнить их. Нужно знать, как они выглядят и как называются. Это нетрудно.

В этой и предыдущих главах мы познакомились с «золотой десяткой» графических фигур ДРАКОНа.

Проверьте себя. Закройте названия на рис. 35 рукой или газетой и, глядя на фигуры, постараитесь вспомнить, как они называются. После этого просмотрите предыдущие рисунки (рис. 1 – 33) и найдите на них все 10 фигур.

Учтите, что в состав золотой десятки входят восемь икон (1-8) и две макроиконы (9 и 10).

Самые ходовые среди них — иконы Действие и Вопрос.

## ВЫВОДЫ

1. Дракон-алгоритм имеет одно начало и один конец.
2. Запрещено иметь в алгоритме несколько концов.
3. Маршрут — путь, идущий из иконы Заголовок до иконы Конец.
4. Главный маршрут — это наиболее благоприятный для пациента путь, соединяющий иконы Заголовок и Конец.
5. Шампур — вертикальная линия между Заголовком и Концом.
6. Шампур изображают жирной линией.

Золотая десятка языка ДРАКОН		
1		Заголовок
2		Конец
3		Действие
4		Вопрос
5		Выбор
6		Вариант
7		Вставка
8		Комментарий
9		Развилка
10		Переключатель

Рис. 35. Часто используемые фигуры в медицинских алгоритмах

7. Главный маршрут алгоритма должен идти по шампуру.
8. Боковой маршрут — любой путь разветвленного алгоритма за исключением главного.
9. Боковые маршруты нужно рисовать справа от шампуря по принципу: «Чем правее, тем хуже».
10. Пересечения линий запрещены.
11. Картографический принцип языка ДРАКОН делает движения взора (по горизонтали и вертикали) осмысленными. Этот принцип вносит в дракон-алгоритм строгую дисциплину и визуальный порядок.
12. Икона Вставка содержит название алгоритма и сообщает, что алгоритм с таким названием нарисован в другом месте.

## Глава 8

# ЛОГИКА В МЕДИЦИНЕ И НЕВИДИМАЯ МАТЕМАТИКА

### КАК ПРЕВРАТИТЬ МЕДИЦИНСКИЙ ТЕКСТ В АЛГОРИТМ? НАДО УБРАТЬ ВСЕ ЛИШНЕЕ

В медицинской литературе алгоритмы обычно существуют в виде «вкраплений», которые перемешаны с другой информацией. Чтобы вычленить алгоритмические знания, их необходимо очистить и отделить от других тем и материалов. Грубо говоря, надо отделить алгоритмическое зерно от шелухи и мякины.

В данном случае «другие темы» (шелуха и мякина) — это различные декларативные медицинские знания, пояснения, обоснования, аргументация, доказательства, ссылки на научные эксперименты и т. д. Это, конечно, совсем не «мякина», а исключительно ценные и важные сведения, но (!) совершенно бесполезные и даже вредные при рисовании алгоритмов. Поэтому при разработке медицинских алгоритмов все лишнее необходимо выделить, отсортировать и удалить.

Поясним сказанное на простом примере, относящемся к аутоиммунной гемолитической анемии.

«При резистентности к кортикоステроидам и внутривенному введению иммуноглобулина в последние годы ряд авторов рекомендуют применять ритуксимаб (мабтера) — химерные человеческие моноклональные антитела против CD20» [135].

Удалив из цитаты «шелуху и мякину», получим:

«При резистентности к кортикостеоидам и внутривенному введению иммуноглобулина... рекомендуют применять ритуксимаб...» [135].

Это и есть искомое алгоритмическое «вкрапление», которое нас интересует.

## КАК ПРЕВРАТИТЬ АЛГОРИТМИЧЕСКИЙ ТЕКСТ В ДРАКОН-АЛГОРИТМ

Мы сделали первый ход и получили словесное описание алгоритма. Теперь надо преобразовать его в графику. Это можно сделать двумя способами, как показано на рис. 36 и 37.

Подчеркнем: на обоих рисунках изображен один и тот же алгоритм. Он лишь нарисован по-разному. На первом рисунке выход Нет слева, на втором справа. Однако эта разница никак не влияет работу алгоритма.

Два алгоритма называются *равносильными*, если они имеют в точности одинаковые маршруты. Отсюда следует, что алгоритмы на рис. 36 и 37 равносильны.



Рис. 36. Алгоритм лечения со сложным условием.  
Выход Нет слева



Рис. 37. Алгоритм лечения со сложным условием.  
Выход Нет справа

## В ДРАКОНе ЗАПРЕЩЕНЫ СЛОЖНЫЕ УСЛОВИЯ. ЧТО БУДЕМ ДЕЛАТЬ?

На рис. 36 и 37 в иконе Вопрос записано сложное логическое условие. Мы уже знаем, что так делать нельзя. Необходимо исправить ошибку, убрать сложное условие и заменить его на два простых. Результат показан на рис. 38 и 39.

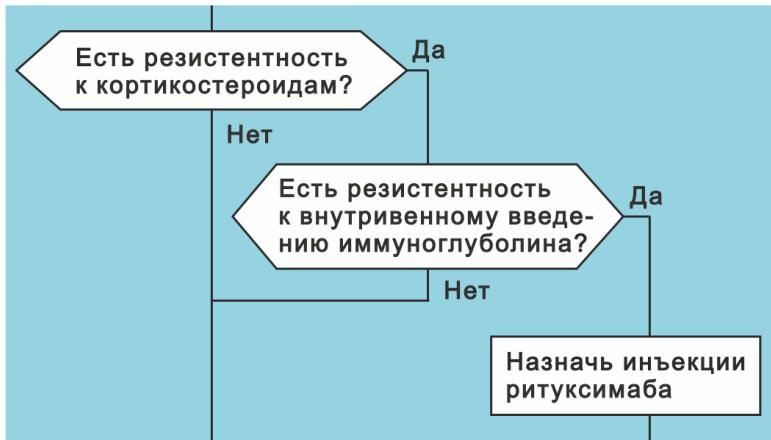
На первый взгляд, между рисунками нет сходства. Однако это не так. Легко убедиться, что на рис. 38 и 39 присутствуют три совершенно одинаковых маршрута. Можно математически доказать, что алгоритмы на этих рисунках в точности совпадают, они равносильны<sup>10</sup>.

Итак, мы выяснили, что два рисунка алгоритмически равны, но отличаются графически. Отличие связано со зрительным восприятием и эргономикой — левый рисунок эргономичнее и удобнее. Причина в том, что слева (на рис. 38) соблюдается картографический принцип ДРАКОНа, а справа — нет.

Поясним. Левая вертикаль на рис. 38 описывает более благоприятный для пациента маршрут (поскольку резистентность к кортикоэстерионам отсутствует), а правая — менее благоприятный (приходится делать инъекции ритуксимаба). Следовательно, при движении взора слева направо выполняется правило: «Чем правее, тем хуже».

Картографический принцип обеспечивает эргономическую упорядоченность и концептуальное единство во всех частях дракон-алгоритма.

<sup>10</sup> Доказательство см. в работе [195].



**Рис. 38.** Алгоритм лечения с двумя простыми условиями. Выход Нет слева.  
Картографический принцип соблюдается [135].



**Рис. 39.** Алгоритм лечения с двумя простыми условиями. Выход Нет справа.  
Картографический принцип нарушен.

## КАК ВЫЯВИТЬ ЛОГИЧЕСКИЕ ПРИНЦИПЫ. ОБСУЖДЕНИЕ МЕТОДИКИ

Наша цель в этой главе — выявить Логику (логические принципы) РАЗВЕТВЛЕННЫХ медицинских алгоритмов, содержащих СЛОЖНЫЕ логические условия. Чтобы решить задачу, мы избрали следующий путь.

1. Найти в медицинской литературе подходящий пример в виде одного или двух предложений, которые содержат алгоритмическое «вкрапление», причем «вкрапление» должно иметь в своем составе:
  - разветвленный алгоритм,
  - сложное логическое условие.
2. Освободить выбранное предложение от посторонней информации, убрать лишние слова и вычленить алгоритмическую часть в чистом виде.
3. Преобразовать алгоритм из текстовой формы в пару равносильных графических алгоритмов. В обоих алгоритмах сложное логическое условие следует поместить в икону Вопрос, как на рис. 36, 37.
4. Преобразовать сложное логическое условие в несколько простых условий, причем каждое простое условие следует записать в отдельной иконе Вопрос (как на рис. 38, 39).
5. Выбрать из пары равносильных алгоритмов один, который удовлетворяет картографическому принципу языка ДРАКОН и признать его **эргономичным и удобным для работы**. А второй алгоритм забраковать и отбросить.

Пример, описанный в начале главы, полностью соответствует данной методике. Ниже представлены еще несколько иллюстраций.

## ПРИМЕР 2. КАК ПРЕВРАТИТЬ МЕДИЦИНСКИЙ ТЕКСТ В ЭРГОНОМИЧНЫЙ АЛГОРИТМ

Рассмотрим еще одно алгоритмическое «вкрапление», относящееся к хронической обструктивной болезни легких (ХОБЛ).

«Ингаляционные глюкокортикоиды и ингаляционные глюкокортикоиды в комбинации с бронхолитиками длительного

действия назначаются при тяжелом течении ХОБЛ, при частых обострениях ХОБЛ — с целью снижения частоты обострений и улучшения качества жизни больных ХОБЛ» [136].

Слова «с целью снижения частоты обострений...» являются хорошим пояснением, но в алгоритме они неуместны. Алгоритм не научный трактат, здесь нельзя растекаться мыслью по древу. В алгоритмах полагается каленым железом выжигать шелуху и мякину.

Удалив все ненужное, получим:

«Ингаляционные глюкокортикоиды и ингаляционные глюкокортикоиды в комбинации с бронхолитиками длительного действия назначаются при тяжелом течении ХОБЛ, при частых обострениях ХОБЛ» [136].

Преобразуем последний текст в графику. Как и раньше, сделаем это двумя способами и получим пару равносильных алгоритмов (рис. 40 и 41).



**Рис. 40.** Алгоритм ХОБЛ со сложным условием.  
Выход Нет слева



**Рис. 41.** Алгоритм ХОБЛ со сложным условием.  
Выход Нет справа

Согласно правилам ДРАКОНа заменим сложное условие на простое. Два искомых равносильных алгоритма показаны на рис. 42 и 43. Первый является хорошим и эргономичным (карографическим). А второй плохим, неудобным и ненужным.



**Рис. 42.** Алгоритм ХОБЛ с двумя простыми условиями. Выходы Нет слева. Картографический принцип соблюдается [136].



**Рис. 43.** Алгоритм ХОБЛ с двумя простыми условиями. Выход Нет справа. Картографический принцип нарушен.

## ПРИМЕР 3. КАК ПРЕВРАТИТЬ СЛОЖНЫЙ МЕДИЦИНСКИЙ ТЕКСТ В АЛГОРИТМ

Рассмотрим более сложную алгоритмическую логику, связанную с транзиторными ишемическими атаками. Как всегда, начнем с цитаты:

«Выявление клинических симптомов транзиторных ишемических атак (онемение лица, половины тела, руки, ноги, кратковременные эпизоды выпадения зрения, эпизоды онемения языка, эпизоды головокружений), нарастающей церебральной недостаточности (снижения работоспособности, появление нарушений речи, снижение качества мыслительных процессов, другие когнитивные нарушения), появление шума в проекции сонных артерий при аусcultации шеи, должно быть основанием для направления больного для проведения ультразвукового допплерографического исследования состояния кровотока магистральных артерий головы. При выявлении стеноза пациент должен быть (!!) направлен к сосудистому хирургу для решения вопроса о тактике оперативного или медикаментозного лечения» [137].

Удалив пояснения и лишние слова, получим концентрированную мысль:

«Выявление клинических симптомов транзиторных ишемических атак..., нарастающей церебральной недостаточности..., появление шума в проекции сонных артерий при аускультации шеи, должно быть основанием для... ультразвукового допплерографического исследования состояния кровотока магистральных артерий головы. При выявлении стеноза пациент должен быть... направлен к сосудистому хирургу для решения вопроса о тактике оперативного или медикаментозного лечения» [137].

Преобразуем последний текст в графику. Действуя, как раньше, по накатанной колее, получим пару равносильных алгоритмов (рис. 40 и 41).

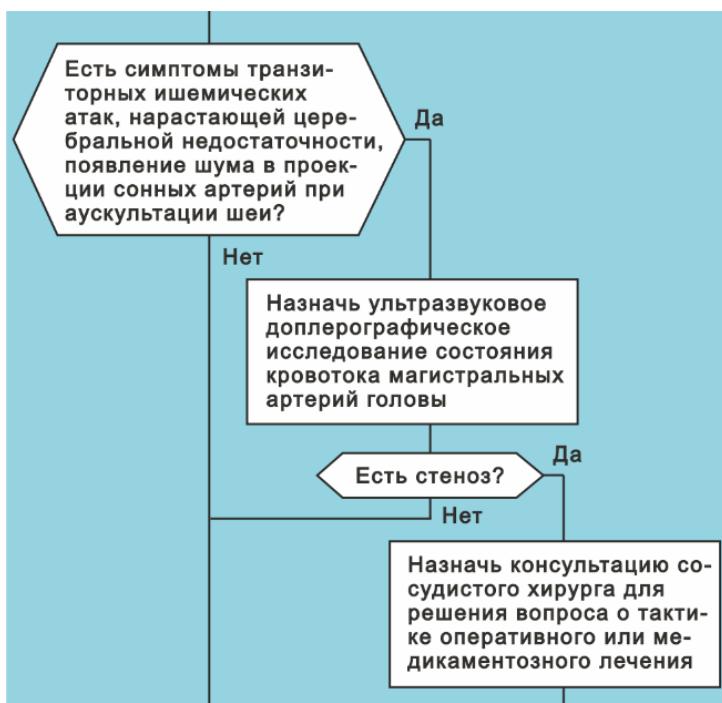
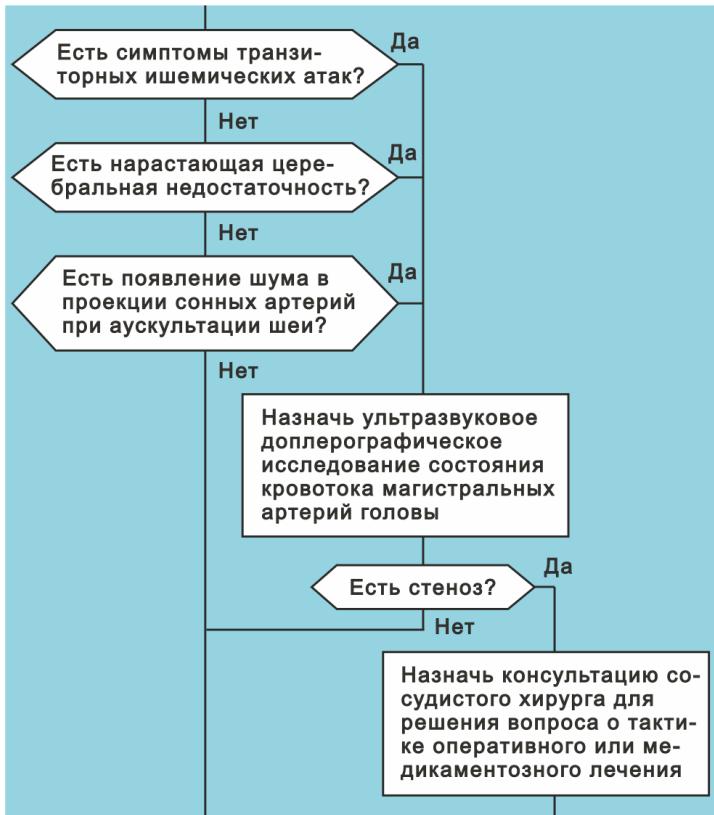


Рис. 44. Алгоритм со сложным условием. Выход Нет слева. Картографический принцип соблюдается.



Рис. 45. Алгоритм со сложным условием. Выход Нет справа. Картографический принцип нарушен.

Сделаем последнее улучшение — разобьем сложное условие на мелкие порции. В итоге увидим два равносильных алгоритма на рис. 42 и 43. Первый является хорошим и упорядоченным (карографичным), второй не удовлетворяет требованиям и отбрасывается.



**Рис. 46.** Хороший алгоритм. Он удовлетворяет всем требованиям [137].



**Рис. 47.** Плохой алгоритм. Требования не выполнены, так как картографический принцип нарушен.

## КАК ПОМОЧЬ СТУДЕНТАМ ИЗУЧАТЬ МЕДИЦИНУ

В предыдущих параграфах мы рассмотрели проблему алгоритмических «вкраплений», со всех сторон окруженных посторонней информацией. Парадокс в том, что окружающая медицинская информация сама по себе очень важна и крайне необходима, но при описаниях медицинских алгоритмов она не нужна и вредит делу.

Алгоритмические «вкрапления» похожи на множество крошечных островков в безбрежном океане медицинской литературы. Между островками должны быть соединительные алгоритмические мостики, но они, как правило, лишь подразумеваются и явном виде не описаны. Это фундаментальный недостаток глобальной медицинской литературы.

Подавляющее число вкраплений существуют в медицинских публикациях в виде изолированных точек. Они остаются одинокими алгоритмическими сиротами, не объединяются в логически законченные фрагменты и не превращаются в целостные алгоритмы.

Разумеется, студенты-медики и врачи в конце концов узнают многие алгоритмические «секреты», используя разные источники и собственный практический опыт. Но такое обучение нерационально и ресурсорасточительно. На подобное обучение люди вынуждены затрачивать неоправданно много времени. Иной

раз почти всю жизнь. Можно ли сэкономить и сократить время, расходуемое на приобретение профессиональных медицинских знаний?

Отсутствие удобных, легко воспринимаемых эргономичных графических алгоритмов высокой точности в учебниках, стандартах, руководствах, клинических рекомендациях, протоколах до добра не доводит. Оно приводит к тому, что тяжкое бремя познания процедурных медицинских знаний взваливается на самих учащихся — на студентов-медиков и слушателей системы последипломного медицинского образования.

Сказанное означает, что проблема алгоритмизации медицинской литературы не только не решена, но даже не осознана.

Акцентируя внимание на недостатках, можно сказать, что современная медицинская литература не упорядочена. Она представляет собой эклектическую смесь (хаотическую мешанину) из декларативных и процедурных знаний.

Первоочередная задача состоит в том, чтобы:

- четко разграничить два типа знания,
- объединить алгоритмические вкрапления с помощью языка ДРАКОН и преобразовать их сначала в законченные алгоритмы, а затем в алгоритмические системы и комплексы.

В данной главе мы делаем лишь самый первый, предварительный шаг и решаем ограниченную задачу. С помощью двенадцати наглядных рисунков (рис. 36 – 47) мы проанализировали ситуацию и показали, что язык ДРАКОН позволяет преобразовать элементарные текстовые «вкрапления» в графическую форму, упорядочить графику с помощью когнитивно-эргономических методов. И за счет этого существенно облегчить восприятие, осмысление и понимание простейших алгоритмов.

На этом мы завершаем анализ проблемы текстовых алгоритмических «вкраплений» и переходим к исследованию и совершенствованию логики графических медицинских алгоритмов.

## КРИТИКА И ИСПРАВЛЕНИЕ БЛОК-СХЕМЫ АЛГОРИТМА

Для наших целей удобно одновременно рассматривать пару равносильных алгоритмов. Продолжим эту традицию.

На рис. 48 и 49 изображены два равносильных алгоритма, нарисованные согласно стандарту на блок-схемы ГОСТ 19.701-90.

С точки зрения когнитивной эргономики, можно отметить ряд недостатков, характерных для блок-схем.

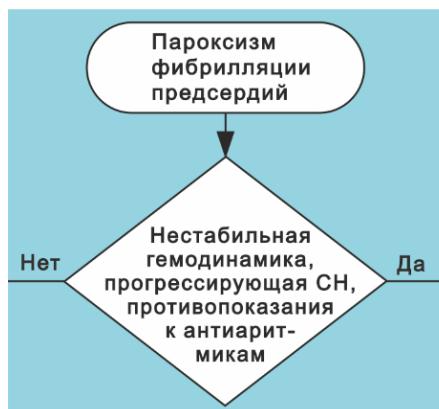


Рис. 48. Ромб и сложное условие. Выход Нет слева

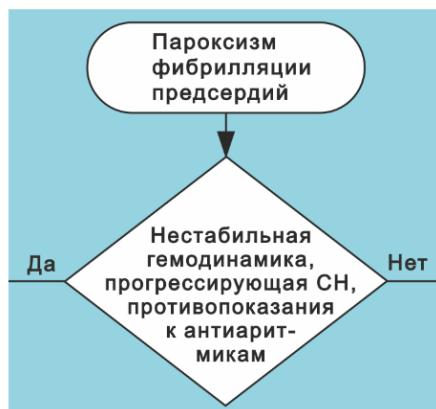


Рис. 49. Ромб и сложное условие. Выход Нет справа

- Для обозначения развилики используется ромб, который занимает слишком много места и содержит мало текста. Ромб не позволяет поместить внутри необходимое количество удобочитаемого текста, состоящего из строк равной длины. Верхний и нижний углы ромба пропадают впустую, так как в них ничего нельзя записать.

В языке ДРАКОН ромбы запрещены, вместо них используется экономичная икона Вопрос. У нее верхний и нижний углы ромба «отпилены». Поэтому схема становится компактной и удобной как для записи текста, так и для чтения.

- Блок-схема не упорядочена. Картографический принцип к ней не применим. Соединительные линии могут быть направлены в любую сторону. По этой причине в блок-схеме, чтобы указать направление движения бегунка по маршруту, необходимо рисовать множество стрелок, которые засоряют схему.

Дракон-схема строго упорядочена, она опирается на картографический принцип. Поэтому отпадает необходимость использовать стрелки (они нужны только в циклах).

## ПРОДОЛЖЕНИЕ КРИТИКИ

Возвратимся к медицине. На рис. 49 изображен фрагмент алгоритма из учебника «Поликлиническая терапия», представленный в виде блок-схемы [138].

Внутри ромба записано трудное для понимания сложное условие, которое является скрытым источником врачебных ошибок. Преобразуем ромб в набор простых условий, причем сделаем это двумя способами — в виде двух равносильных алгоритмов, как показано на рис. 50 и 51.

Сравнивая их, можно заметить, что картографический принцип выполняется лишь слева, на рис. 50. Только этот алгоритм можно признать *эргономичным и удобным для работы*. Почему?

Ответить нетрудно. Все три условия («Есть нестабильная гемодинамика?» и остальные) носят негативный характер, они неблагоприятны для пациента. Чтобы отвести беду от больного, надо устранить негативные моменты и выбрать наилучший для него маршрут.

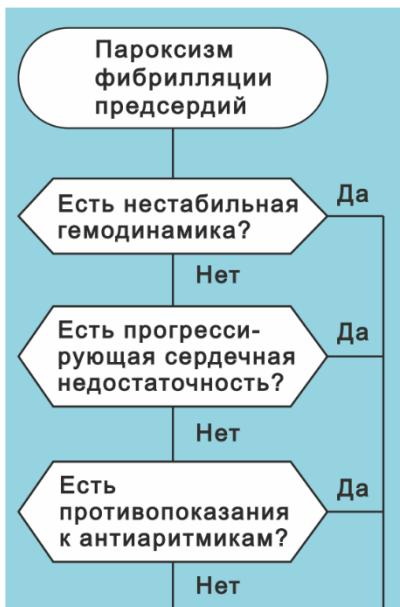


Рис. 50. Картографический принцип соблюдается [138]



Рис. 51. Картографический принцип нарушен

Для этого на все три негативных вопроса следует ответить Нет. Это и будет «хороший» маршрут, описывающий благоприятную для пациента ситуацию. На рис. 50 данный маршрут идет по шампуру, он является главным. Любое отклонение вправо от главного маршрута неблагоприятно и уводит нас на «плохой» маршрут. Следовательно, алгоритм на рис. 50 удовлетворяет правилу ДРАКОНа «Чем правее, тем хуже». Алгоритм на рис. 51 таким свойством не обладает.

## НЕГАТИВНЫЕ И ПОЗИТИВНЫЕ ВОПРОСЫ

В этой главе мы много раз упражнялись, преобразуя сложные условия в простые. Перечислим наши упражнения.

- На рис. 38 и 39 показаны два простых условия (две иконы Вопрос).
- На рис. 42 и 43 также нарисованы две иконы Вопрос.
- На рис. 46 и 47 изображены уже не два, а три простых условия (три иконы Вопрос).
- На рис. 50 и 51 — тоже три иконы Вопрос.

А теперь самое любопытное. Оказывается, во всех случаях мы думали о плохом, а не о хорошем.

Мы нарочно использовали только *негативные* вопросы, т. е. вопросы, которые при ответе Да, указывают на неблагоприятную для пациента ситуацию. Например, «Есть прогрессирующая сердечная недостаточность?». Если есть, это плохо для пациента.

Пришло время перейти к анализу *позитивных* вопросов. Позитивный вопрос при ответе Да указывает на благоприятное для больного положение дел.

## ЛОГИЧЕСКАЯ СХЕМА «ИЛИ»

Рассмотрим рис. 52 и 53. Пациент находится в угрожающем для жизни состоянии, решается вопрос о реанимации. В первую очередь врач должен ответить на вопросы:

- Есть ли реакция пациента на обращение и прикосновение?
- Есть ли нормальное (не агональное) дыхание?

Если оба ответа «Нет», нужна реанимация, в противном случае — тщательный первичный осмотр.

С точки зрения медицины, реанимация — это главное. Однако у нас на уме другое — нас интересует логическая операция «ИЛИ». По этой причине забудем про реанимацию и переключим внимание на другое.

Попытаемся понять: в каком случае нужно проводить первичный осмотр пациента? Алгоритм на рис. 52 и 53 говорит, что это возможно в трех случаях.

1. Если есть реакция пациента на обращение и прикосновение. (*Маршрут выходит из верхней иконы Вопрос через Да*).
2. Если есть нормальное (не агональное) дыхание. (*Маршрут выходит из верхней иконы Вопрос через Нет, затем пробегает через нижнюю икону Вопрос через Да*).

Это означает, что пациент не реагирует на обращение и прикосновение, однако дышит он normally.

3. Если выполняются оба указанных условия.

В этом случае состояние пациента благоприятное. Во-первых, он реагирует на обращение и прикосновение, во-вторых, у него normalное дыхание.

Перечисленные соображения говорят о том, первичный осмотр пациента в алгоритме на рис. 52 и 53 выполняется по схеме «ИЛИ».

Используя термин «ИЛИ», ситуацию можно описать так. Первичный осмотр следует проводить в следующих случаях:

- ИЛИ пациент реагирует на обращение и прикосновение,
- ИЛИ пациент нормально дышит,
- ИЛИ имеют место оба этих признака.

Оба алгоритма на рис. 52 и 53 являются логически правильными. Оба реализуют схему ИЛИ. Разница лишь в том, что левый алгоритм соблюдает эргономические правила ДРАКОНа (*Главный маршрут должен идти по шампуру. Чем правее, тем хуже*). И это хорошо.

А правый не соблюдает. И это плохо.

И последнее. На рис. 52 и 53 в иконах Вопрос мы впервые использовали позитивные вопросы. Этим они отличаются от рисунков 36 – 51, где повсеместно использовались негативные вопросы.



**Рис. 52.** Схема «ИЛИ» с двумя условиями. Выход Да слева. Картографический принцип соблюдается.



**Рис. 53.** Схема ИЛИ» с двумя условиями. Выход Да справа. Картографический принцип нарушен.

Как создать схему ИЛИ

- Соедините выходы «Да» двух икон Вопрос.
- Выход «Нет» верхней иконы Вопрос соедините со входом нижней иконы Вопрос.
- Объединенный выход «Да» двух икон Вопрос выполняет операцию ИЛИ.

## ЛОГИЧЕСКАЯ СХЕМА «ИЛИ» С ТРЕМЯ УСЛОВИЯМИ

Мы изучили схему «ИЛИ» с двумя условиями (с двумя иконами Вопрос). Рассмотрим такую же схему, но с тремя условиями. Возьмем за основу алгоритм на рис. 52, 53 и внесем в него нужные исправления.

Рассмотрим фразу «Есть ли реакция пациента на обращение и прикосновение?». При желании ее можно разделить на два вопроса. В некоторых случаях это бывает полезно. С учетом этого, взглянем на рис. 54, 55.

Мы видим, что врач должен ответить уже не на два, а на три отдельных вопроса:

- Есть ли реакция пациента на обращение?
- Есть ли реакция на прикосновение?
- Есть ли нормальное (не агональное) дыхание?

Если хотя бы на один вопрос получен ответ «Да», следует выполнить первичный осмотр (в противном случае нужна реанимация).

Обе схемы (рис. 54 и 55) логически правильны и равны друг другу. Какую из них следует предпочесть? Ту, которая удовлетворяет эргономичным правилам ДРАКОНа. В данном случае это левая схема ИЛИ. Правая схема нарушает требование «Чем правее, тем хуже» и отбрасывается.



**Рис. 54.** Схема «ИЛИ» с тремя условиями. Выход Да слева. Картографический принцип соблюдается [159].



**Рис. 55.** Схема «ИЛИ» с тремя условиями. Выход Да справа. Картографический принцип нарушен.

Логическая операция ИЛИ означает: если хотя бы на один вопрос получен ответ «Да», выполнни указанное в алгоритме действие (Первичный осмотр). Оба алгоритма (на рис. 54 и 55) равносильны и отличаются только способом начертания.

## СХЕМА «ИЛИ» ДЛЯ ПОЗИТИВНЫХ И НЕГАТИВНЫХ ВОПРОСОВ

Мы познакомились с двумя новыми понятиями или терминами:

- позитивный вопрос,
- негативный вопрос.

Им соответствуют две разные схемы:

- стандартная схема ИЛИ,
- нестандартная схема ИЛИ.

## СТАНДАРТНАЯ СХЕМА «ИЛИ»

Её следует использовать для позитивных вопросов



Рис. 56. Стандартная схема «ИЛИ» с двумя позитивными вопросами [159]

## НЕСТАНДАРТНАЯ СХЕМА «ИЛИ»

Её следует использовать для негативных вопросов

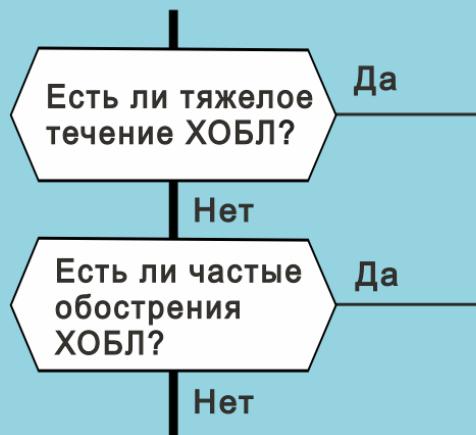


Рис. 57. Нестандартная схема «ИЛИ» с двумя негативными вопросами [136]

## СТАНДАРТНАЯ СХЕМА «ИЛИ»

Её следует использовать для позитивных вопросов

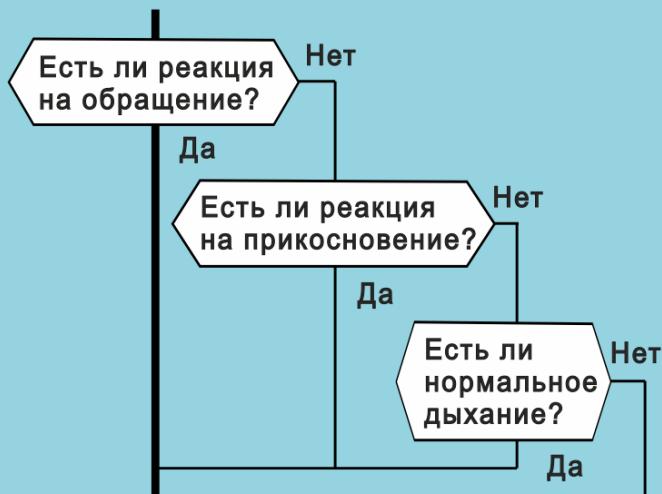


Рис. 58. Стандартная схема «ИЛИ» с тремя позитивными вопросами [159]

## НЕСТАНДАРТНАЯ СХЕМА «ИЛИ»

Её следует использовать для негативных вопросов



Рис. 59. Нестандартная схема «ИЛИ» с тремя негативными вопросами [137]

Все четыре примера на рис. 56–59 удовлетворяют правилам ДРАКОНа и являются правильными. Они рекомендуются в качестве образцов для подражания. Схема ИЛИ соответствует выходу Да.

На рис. 56 и 58 (где позитивные вопросы) шампур идет через Да. На рис. 57 и 59 (где негативные вопросы) шампур идет через Нет.

Стандартная схема ИЛИ похожа на «Лестницу с уступами», нестандартная — на «Мачту с парусами».

На рис. 56 лестница имеет два уступа, на рис. 58 — три. Точно так же на рис. 57 мачта имеет два паруса, на рис. 59 — три.

А если добавить еще один «этаж»? Лестница будет расти в ширину, а мачта — в высоту.

Правило 1  
для схемы ИЛИ

Стандартная схема ИЛИ (лестница) предназначена для изображения двух или более **позитивных** вопросов. При этом шампур помечают словом Да.

Правило 2  
для схемы ИЛИ

Нестандартная схема ИЛИ (мачта) предназначена для изображения двух или более **негативных** вопросов. При этом шампур помечают словом Нет.

## ЛОГИЧЕСКАЯ СХЕМА «И» С ДВУМЯ УСЛОВИЯМИ

Схема И отличается тем, что необходимо совпадение двух или нескольких условий.

Предположим, надо повесить на стену картину. Для этого придется забить в стену гвоздь. Что для этого нужно? Нужен молоток и гвоздь. Если есть только гвоздь, а молотка нет, ничего не выйдет. И наоборот, если есть молоток без гвоздя, толку не будет. Непременно нужно и то и другое. Говорят, что нужно совпадение двух условий: и гвоздя, и молотка (рис. 60 и 61).

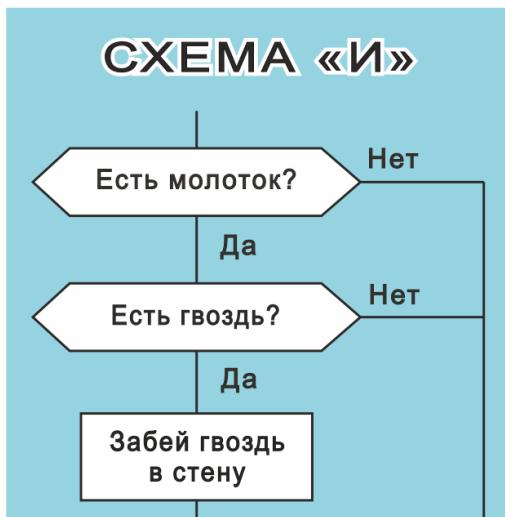


Рис. 60. Схема «И» с двумя условиями. Выход Да слева. Картографический принцип соблюдается.

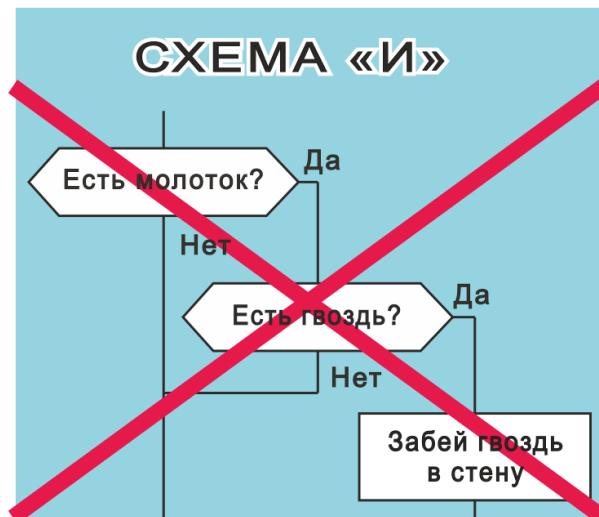


Рис. 61. Схема «И» с двумя условиями. Выход Да справа. Картографический принцип нарушен.

Логическая операция И означает: если на оба вопроса получен ответ «Да», выполнить действие, на которое указывает выход Да нижней иконы Вопрос (Забей гвоздь в стену). Оба алгоритма (на рис. 60 и 61) равносильны и отличаются только способом начертания.

## ЛОГИЧЕСКАЯ СХЕМА «И». МЕДИЦИНСКИЙ ПРИМЕР

Рассмотрим равносильные алгоритмы на рис. 62 и 63. Речь идет об интубации трахеи. Предлагаются два вопроса для самоконтроля врача:

- Можешь ли провести интубацию?
- Больше ли пользы в интубации, чем риска?

Если оба ответа «Да», врач приступает к интубации.

Алгоритм на рис. 62 и 63 говорит, что проведение интубации трахеи возможно при совпадении двух условий.

1. Если врач обладает достаточным опытом и способен провести интубацию. (Маршрут выходит из верхней иконы Вопрос через Да).
2. Если врач уверен, что польза от интубации превышает возможный риск для пациента. (Маршрут выходит из нижней иконы Вопрос через Да).

Эти соображения говорят о том, интубация трахеи в алгоритме на рис. 62 и 63 выполняется по схеме «И».

Используя термин «И», ситуацию можно описать так. Интубацию трахеи следует проводить, если одновременно выполняются два условия:

- врач способен провести интубацию И
- польза от интубации больше, чем риск.

Оба алгоритма на рис. 62 и 63 являются логически правильными. Оба реализуют схему И. Разница лишь в том, что левый алгоритм соблюдает эргономические правила ДРАКОНа (Главный маршрут идет по шампуру. Чем правее, тем хуже).

А правый не соблюдает. Поэтому правый алгоритм считается негодным и отбрасывается.

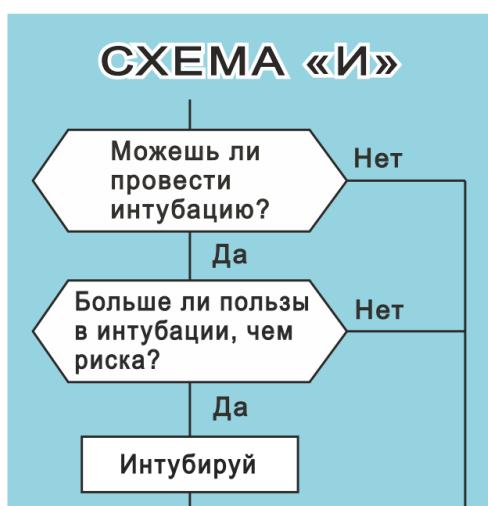


Рис. 62. Схема «И» с двумя условиями. Выход Да слева. Картографический принцип соблюдается [196].

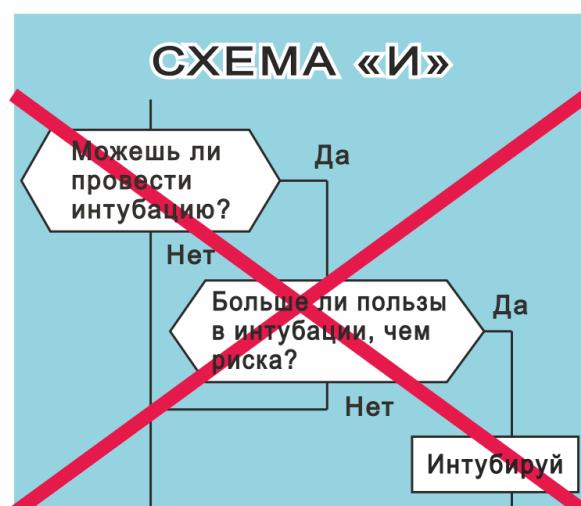


Рис. 63. Схема «И» с двумя условиями. Выход Да справа. Картографический принцип нарушен.

Как создать схему «И»

- Соедините выход «Да» верхней иконы Вопрос со входом нижней иконы Вопрос.
- Соедините выходы «Нет» обеих икон Вопрос.
- Выход «Да» нижней иконы Вопрос выполняет операцию «И».

## ЛОГИЧЕСКАЯ СХЕМА «И» С ТРЕМЯ УСЛОВИЯМИ

Мы изучили схему «И» с двумя условиями (с двумя иконами Вопрос). Рассмотрим более сложную схему на рис. 64, 65, которая содержит не два, а три условия.

Врач должен получить ответ на три вопроса:

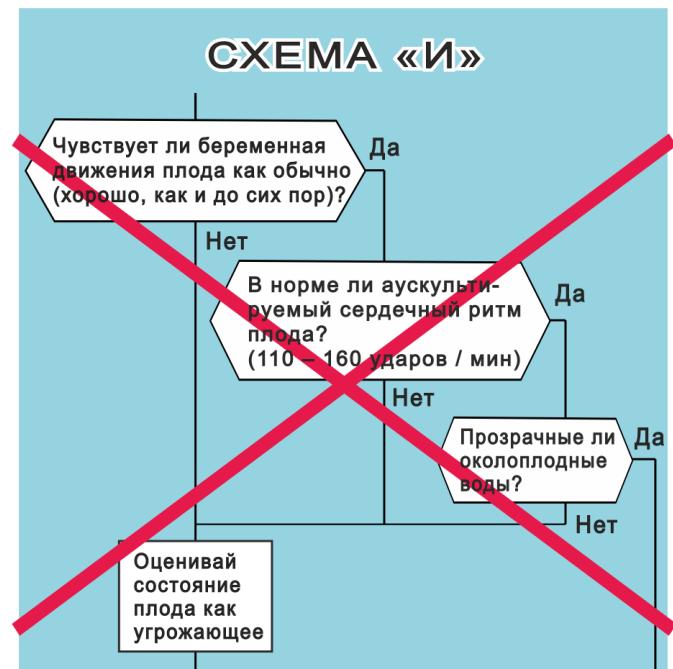
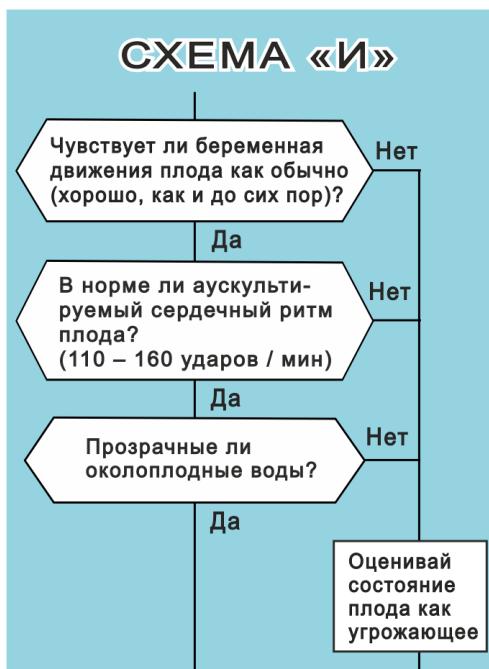
- Чувствует ли беременная женщина движения плода, как обычно?
- В норме ли аускультируемый сердечный ритм плода?
- Прозрачны ли околоплодные воды?

Если на все вопросы получен ответ «Да», это благоприятный исход.

Если же хотя бы на один вопрос ответ «Нет», состояние плода следует оценить, как угрожающее.

Оба алгоритма (рис. 64 и 65) делают одно и то же. Какой из них следует предпочесть? Ответ: левый, так как в нем выполняется эргономичное правило «Чем правее, тем хуже».

Угрожающее состояние плода — это худший вариант. Следовательно, правый алгоритм нарушает требования и бракуется.



## СХЕМА «И» ДЛЯ ПОЗИТИВНЫХ И НЕГАТИВНЫХ ВОПРОСОВ

Вернемся еще раз к необходимости различать два понятия:

- позитивный вопрос,
- негативный вопрос.

Им соответствуют две разные схемы:

- стандартная схема И,
- нестандартная схема И.

## СТАНДАРТНАЯ СХЕМА «И»

Её следует использовать для позитивных вопросов



Рис. 66. Стандартная схема «И» с двумя позитивными вопросами [196]

## НЕСТАНДАРТНАЯ СХЕМА «И»

Её следует использовать для негативных вопросов



Рис. 67. Нестандартная схема «И» с двумя негативными вопросами [189]

## СТАНДАРТНАЯ СХЕМА «И»

Её следует использовать для позитивных вопросов

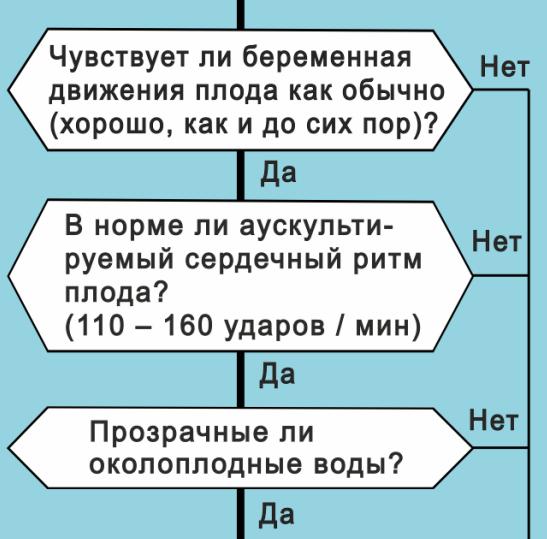


Рис. 68. Стандартная схема «И» с тремя позитивными вопросами [197]

## НЕСТАНДАРТНАЯ СХЕМА «И»

Её следует использовать для негативных вопросов

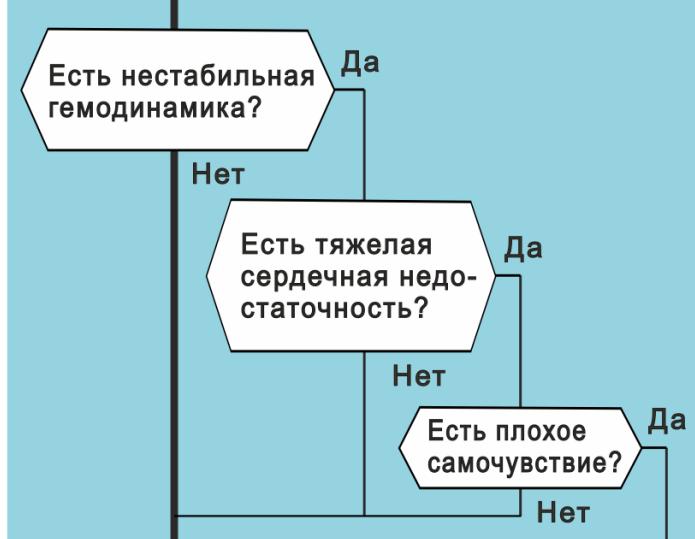


Рис. 69. Нестандартная схема «И» с тремя негативными вопросами [138]

Все четыре примера на рис. 66–69 удовлетворяют правилам ДРАКОНа и являются правильными. Они рекомендуются в качестве образцов для подражания. Схема И соответствует выходу Да.

На рис. 66 и 68 (где позитивные вопросы) шампур идет через Да, на рис. 67 и 69 (негативные вопросы) шампур идет через Нет.

## МНЕМОНИЧЕСКОЕ ПРАВИЛО

Стандартная схема И похожа на «Мачту с парусами», нестандартная — на «Лестницу с уступами».

На рис. 66 мачта имеет два паруса, на рис. 68 — три. Точно так же на рис. 67 лестница имеет два уступа, на рис. 69 — три.

Если добавить еще один или несколько «этажей», мачта будет расти в высоту, а лестница — в ширину.

Правило 1  
для схемы И

Стандартная схема И (мачта) предназначена для изображения двух и более **позитивных** вопросов. При этом шампур помечают словом Да.

Правило 2  
для схемы И

Нестандартная схема И (лестница) предназначена для изображения двух и более **негативных** вопросов. При этом шампур помечают словом Нет.

## НЕВИДИМАЯ МАТЕМАТИКА. СХЕМА «ИЛИ» И ЗАКОН ДЕ МОРГАНА

Мы рассмотрели свыше тридцати примеров с профессиональными медицинскими текстами, которые выполняют различные логические функции (рис. 36 – 69). Во всех случаях логические функции реализованы с помощью «невидимой математики», то есть с помощью графики. Графика языка ДРАКОН хороша тем, что позволяет полностью отказаться от логических математических формул.

Как это делается?

Начнем от печки и освежим в памяти рисунки 56 и 57. Удалим мысленно медицинский текст в иконах Вопрос. Для краткости заменим его одной буквой (A и B). Результат такого «переодевания» изображен на рис. 70.

Поедем дальше. Главный выход схемы ИЛИ обозначим буквой Z. Второй выход всегда является инверсным и обозначается через  $\bar{Z}$ , где  $\bar{Z}$  — логическое отрицание логической переменной Z.

В нижней части рисунка приведены несложные выкладки, которые показывают, что математическое соотношение между главным и инверсным выходами схемы ИЛИ описывается законом Августа де Моргана (Augustus De Morgan) [139].

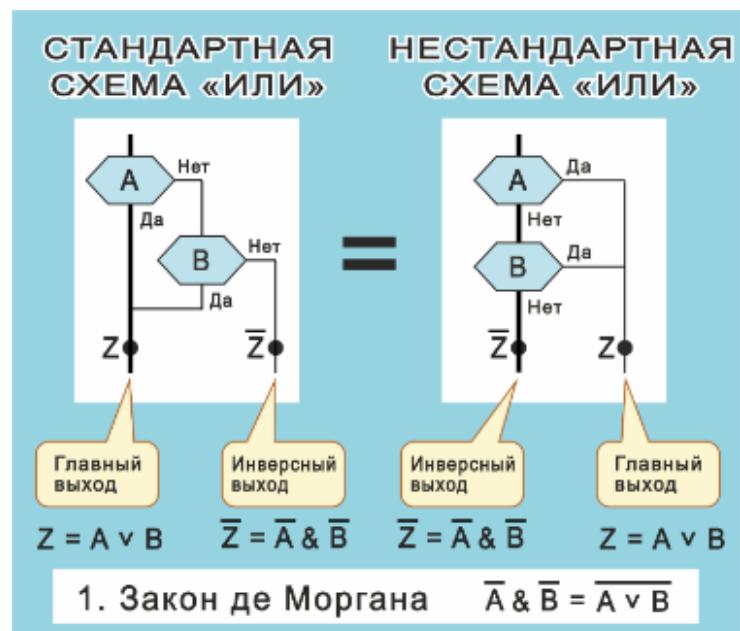


Рис. 70. Две схемы «ИЛИ» в виде лестницы (слева) и мачты (справа) являются равносильными. Они соединены знаком равенства. Связь между главным и инверсным выходами выражает закон де Моргана.

Нужны ли эти выкладки для практикующего врача? Конечно, нет. Мы привели их только для того, чтобы проиллюстрировать суть понятия «невидимая математика».

Графика языка ДРАКОН полностью заменяет подобные выкладки, отображая математическую сущность проблемы визуальными средствами.

Вывод состоит в том, что — вместо утомительной работы с логико-математическими формулами и таблицами истинности — врачу достаточно запомнить два mnemonicических понятия (Мачта и Лестница) и соответствующие им наглядные зрительные образы.

## СХЕМА «И» И ВТОРОЙ ЗАКОН ДЕ МОРГАНА

На рис. 71 та же самая проблема решена для схемы «И».

Для начала посмотрим на рис. 66 и 67. Медицинский текст в иконах Вопрос на этих рисунках для краткости заменим буквами А и В. Главный выход схемы И обозначим буквой Z. Второй (инверсный) выход является логическим отрицанием Z. Обозначим его через  $\bar{Z}$  (Z с чертой).

В нижней части рисунка, как и раньше, приведены логические формулы. Отличие в том, что на этот раз они относятся к схеме «И».

Обратите внимание. Математическое соотношение между главным и инверсным выходами схемы И описывается не первым, а вторым законом Августа де Моргана.

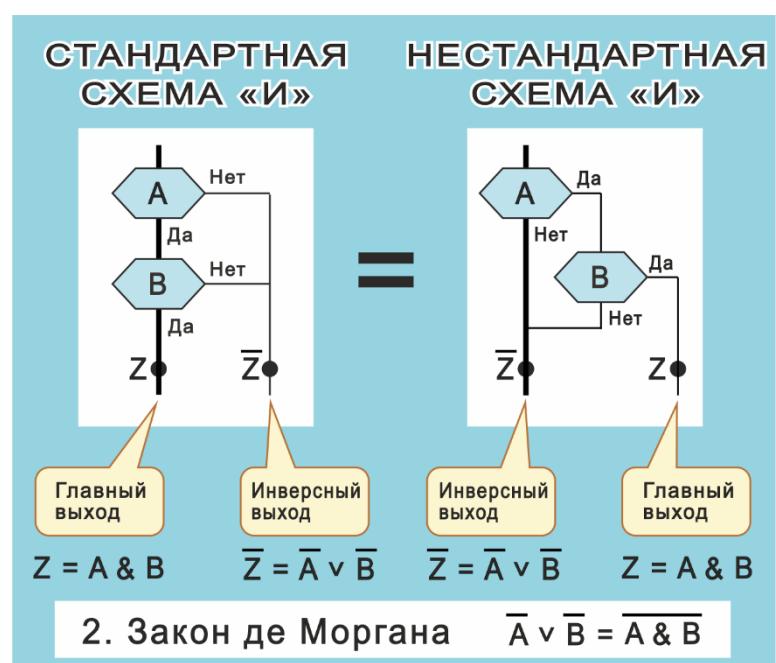


Рис. 71. Две схемы «И» в виде мачты (слева) и лестницы (справа) являются равносильными. Они соединены знаком равенства. Связь между главным и инверсным выходами выражает второй закон де Моргана.

## ЛОГИЧЕСКОЕ ОТРИЦАНИЕ

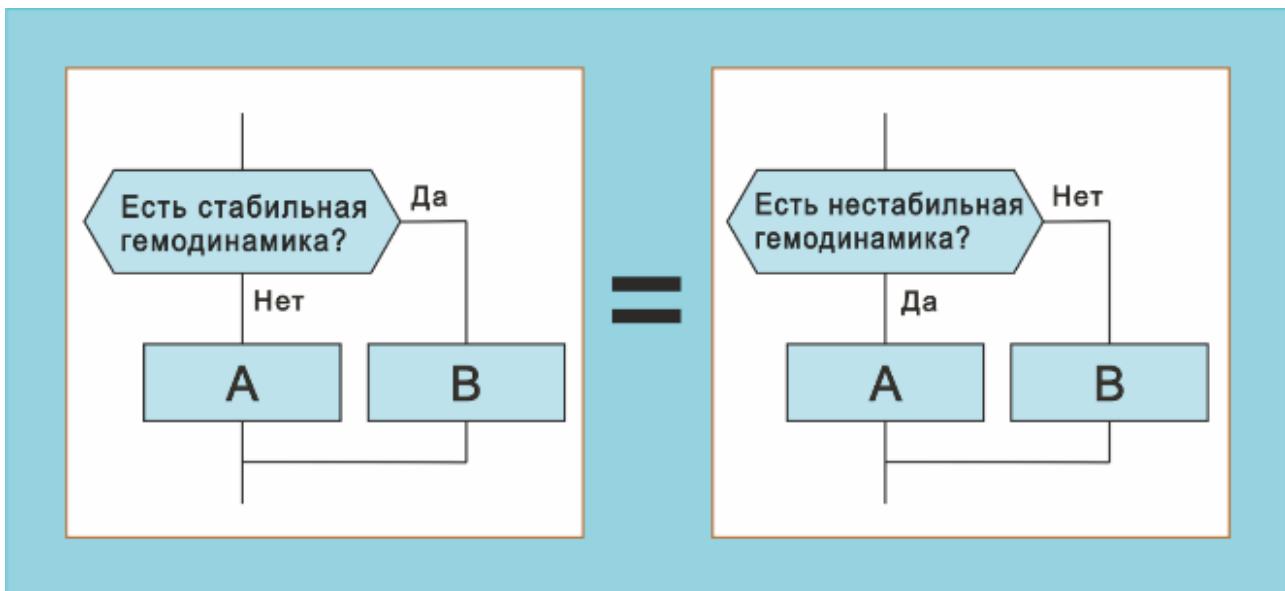
Логическая схема «Не» (логическое отрицание) показана на рис. 72. Уберем отрицание в правой иконе Вопрос — вместо слова «неустойчивая» напишем «стабильная». Одновременно переставим слова Да и Нет. Эти два изменения компенсируют друг друга. В итоге образуется точно такой же (равносильный) алгоритм. Он изображен на рис. 72 слева.

Поясним. Мы применили операцию логического отрицания к алгоритму в правой рамке на рис. 72. В результате логического отрицания получен равносильный алгоритм, показанный в левой рамке.

Два одинаковых алгоритма, как обычно, соединены знаком равенства.

## РОКИРОВКА

Мы приближаемся к самому интересному месту данной главы. Главный секрет связан с понятием *рокировка*.



**Рис. 72.** Пример операции «НЕ» (логическое отрицание). Если в правой развилке удалить отрицательную частицу «не» и поменять местами ответы Да и Нет, получим равносильный алгоритм, нарисованный в левой рамке.

На рисунках 36 – 55 мы брали «каждой твари по паре» и попарно анализировали медицинские алгоритмы. Правый алгоритм из каждой пары мы браковали и зачеркивали жирным крестом. Вместо него в качестве образца выбирали левый алгоритм. И так в каждой паре.

Секрет в том, что забракованный алгоритм мы улучшали, восстанавливали и превращали в красавчика, используя операцию *рокировка*.

Рокировка обладает удивительным свойством. Она может исправить плохое и создать хорошее, переделывая брак в образец. Подобно магическому заклинанию она превращает алгоритмического «урода» в «красавца». Конечно, это возможно лишь тогда, когда оба алгоритма равносильны (рис. 73).

Рокировка — операция, которая видоизменяет внешний облик алгоритма, не меняя его по существу. Это значит, что рокировка — равносильное преобразование алгоритма. При рокировке смысл алгоритма не меняется.

Уточним. Операция «рокировка» относится не ко всему алгоритму, а только к одной развилке (как показано на рис. 73).

Говоря упрощенно, при рокировке левая и правая части развилки меняются местами. Назовем эти части плечами.

*Левое плечо развилки* есть маршрут от нижнего выхода иконы Вопрос до точки слияния. Оно содержит ответ «Да», икону В и вертикальную линию (рис. 73).

*Правое плечо* — путь от правого выхода иконы Вопрос до точки слияния. Оно включает слово «Нет», икону В и соединительные линии.

Что такое рокировка?  
Это плеч перестановка!

## ПРИМЕР РОКИРОВКИ

Пример показан на рис. 74. В чем заключается рокировка в данном случае? Чтобы ответить, надо сравнить развилки в левой и правой рамке. И посмотреть, что изменилось.

Две иконы «Поиграй с кошкой» и «Накорми кошку» поменялись местами. Точно так же переехали слова Да и Нет. Все остальное осталось на прежнем месте.

# РОКИРОВКА

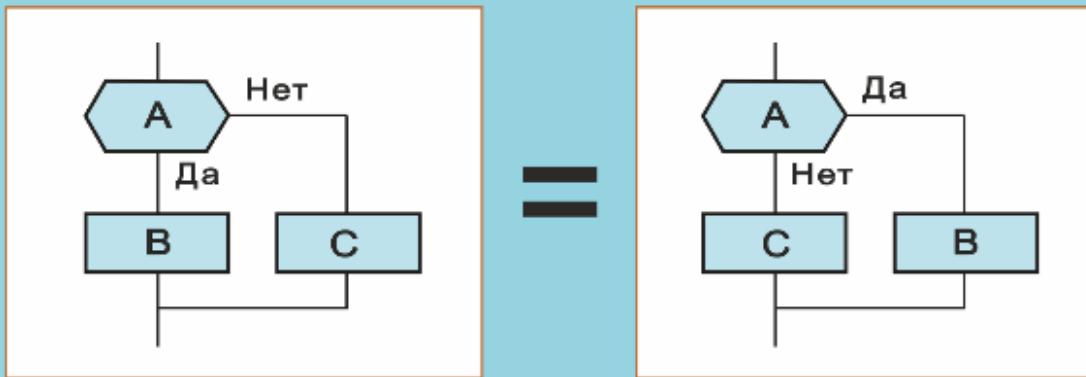


Рис. 73. Рокировка преобразует левую разветвку в правую (можно и наоборот). При этом:

- Графика остается неизменной.
- Меняются местами надписи в иконах Действие.
- Слова Да и Нет также изменяют свои места.

## ЗАЧЕМ НУЖНА РОКИРОВКА

При проведении рокировки следует ответить на два вопроса.

1. Что было и что стало?
2. Зачем нужна рокировка в данном случае?

Предположим, что исходной является схема в правой рамке на рис. 74. Эта схема имеет дефект — нарушено правило «Чем правее, тем хуже». Действительно, голод не тетка. Если кошка голодная, это плохо, если сытая, хорошо.

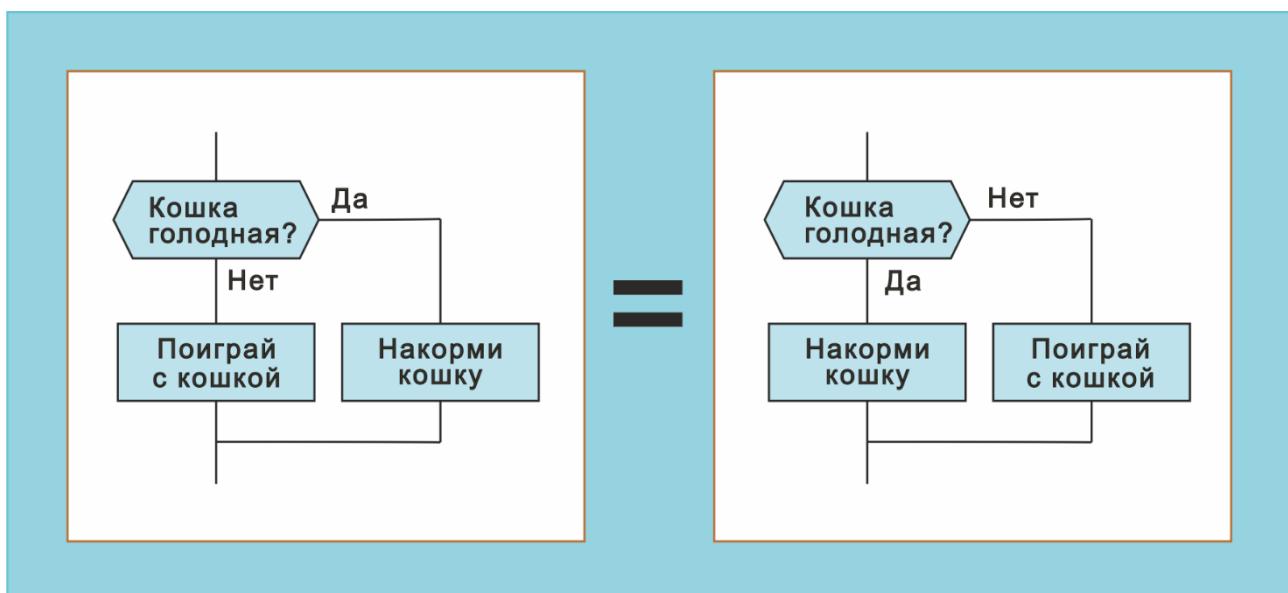
Недостаток правой схемы заключается в том, что голодная кошка находится на главном маршруте. Это неправильно. Чтобы устранить недочет, надо осуществить рокировку. Рокировка перетащит голодную кошку на боковой маршрут, и все будет в порядке.

После рокировки получим схему в левой рамке на рис. 74. Таким образом, плохую схему в правой рамке можно с помощью рокировки улучшить, облагородить и преобразовать в хорошую схему, находящуюся в левой рамке.

Двинемся дальше. Забудем про голодных кошек и свежим глазом глянем на дружные парочки медицинских алгоритмов на рис. 36 – 55.

Отметим, что во всех случаях наблюдается одна и та же ситуация. А именно: в каждой «парочке» исходной является правая, бракованная схема — та, что зачеркнута. Так вот, эту негодную схему необходимо исправить с помощью спасительной рокировки. И превратить в хороший алгоритм, размещенный в левой части каждой пары.

Это означает, что мы сразу получили десяток убедительных примеров (рис. 36, 38, ... 54). Потому что рокировка позволила исправить дефектную схему и превратить ее в хороший медицинский алгоритм.



**Рис. 74.** Пример операции «рокировка». Две схемы являются равносильными. Они соединены знаком равенства. Плохую схему справа можно исправить рокировкой и превратить в хорошую схему слева.

## РОКИРОВКА МОЖЕТ УЛУЧШИТЬ ЭРГОНОМИЧНОСТЬ АЛГОРИТМОВ

Мы старались показать, что рокировка отличная вещь, праздник души, именины сердца. Осталось выяснить, как с пользой для дела использовать ее на практике. Здесь есть заминка — с непривычки можно попасть в лужу.

Чтобы этого не случилось, рассмотрим довольно сложную задачу. Не одну единственную развилку, а большой, солидный алгоритм «Обед в ресторане».

На рис. 75 показана плохая дракон-схема. Согласно правилу главный маршрут (жирная линия) должен быть прямым, как стрела, и идти точно по шампуру. А он вместо этого превратился в ломаную-переломаную линию, которая делает невообразимые скачки и путает читателя. Чтобы исправить ошибку, нужно несколько раз сделать рокировку.

Интересно, сколько раз? Оказывается, шесть! Чтобы вылечить больную схему, необходимо выполнить операцию «рокировка» ровно шесть раз!

Начнем по порядку.

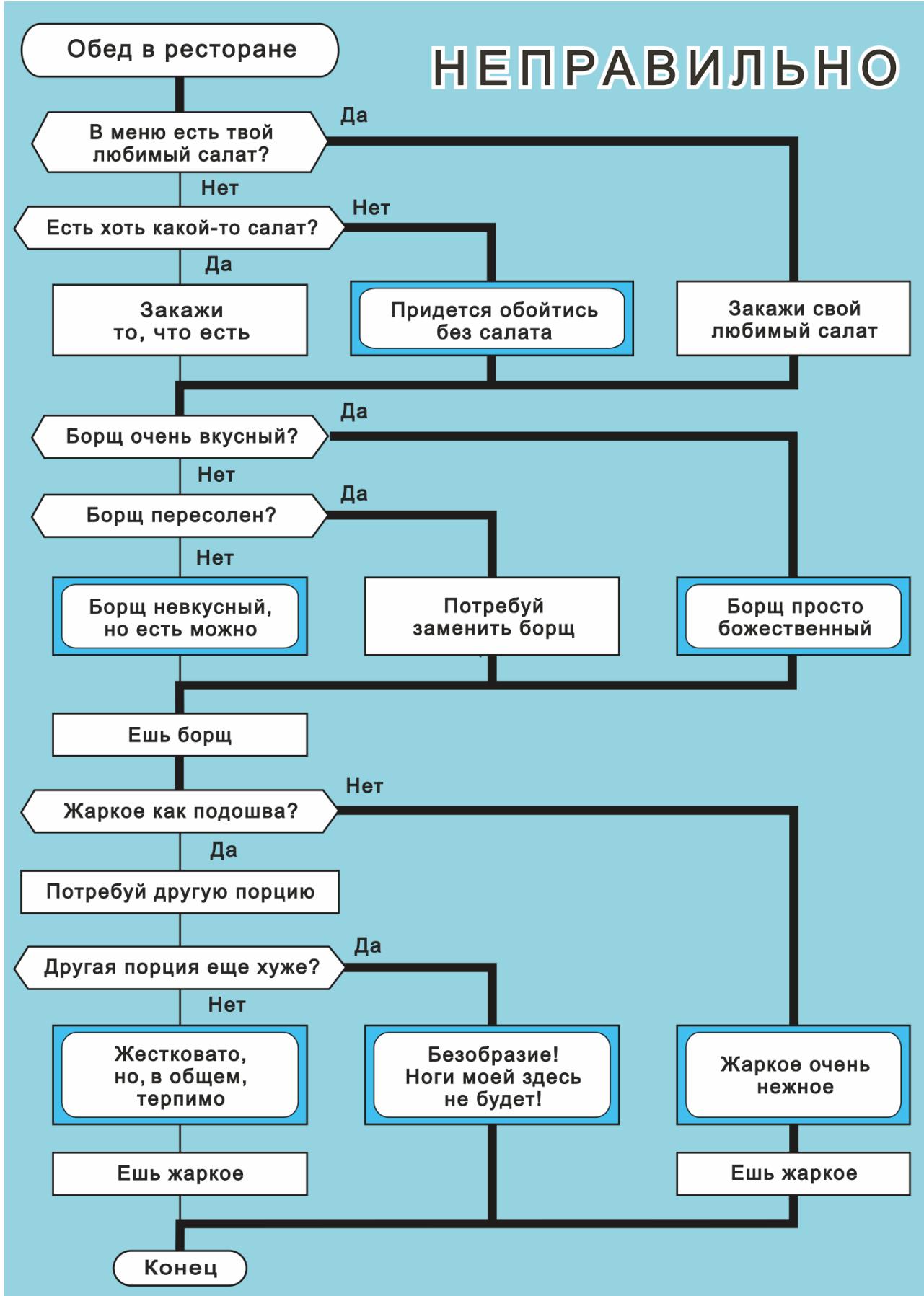
Первый раз делаем рокировку в развилке «В меню есть ваш любимый салат?», второй — в развилке «Есть хоть какой-то салат?». Двойная рокировка позволяет пустить главный маршрут по шампуру на верхнем участке. Однако внизу главный маршрут по-прежнему совершает неоправданные зигзаги.

Затем делаем рокировку в развилке «Борщ очень вкусный?». И еще раз в развилке «Борщ пересолен?». Тем самым улучшаем среднюю часть схемы.

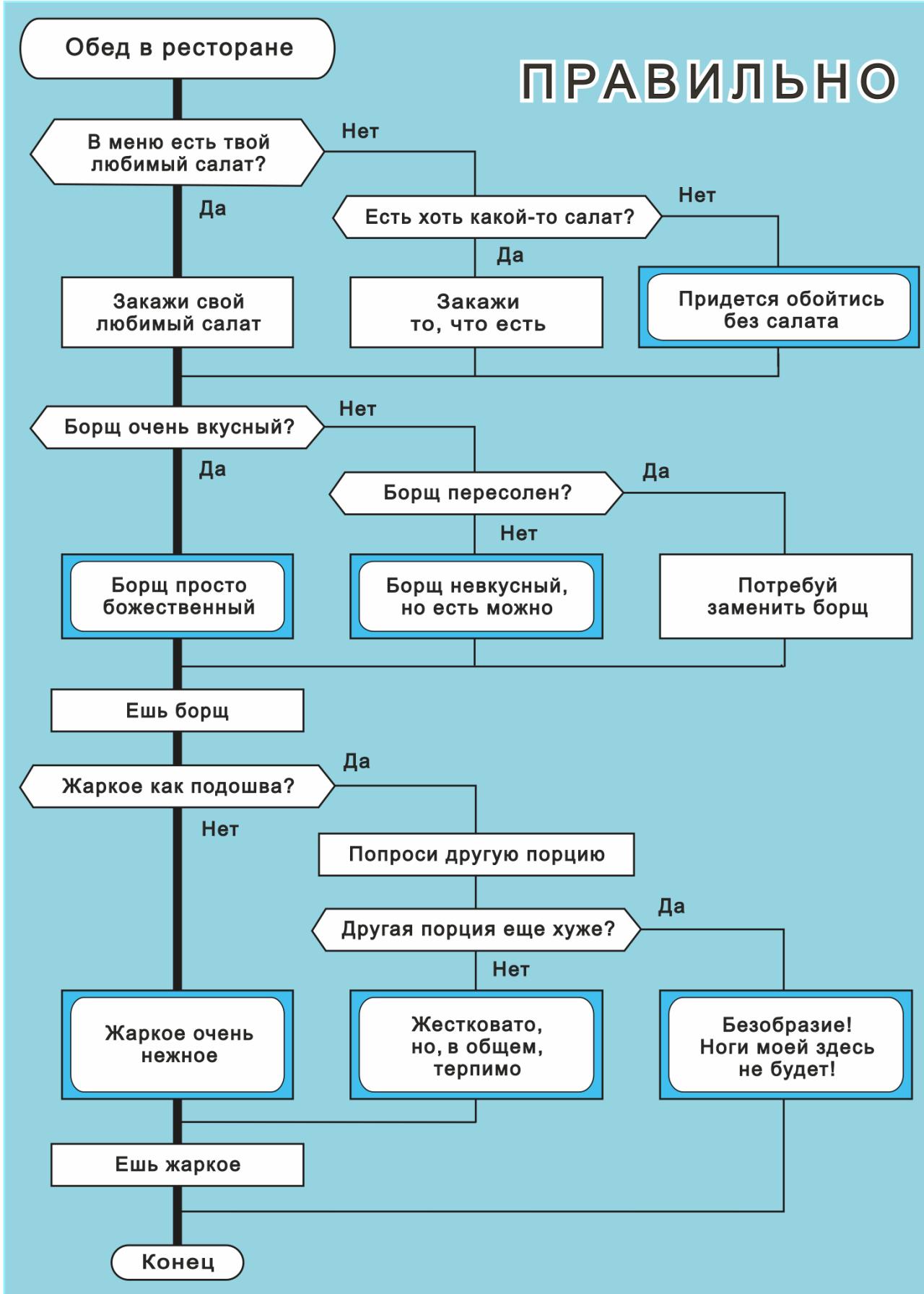
Нам осталось выпрямить главный маршрут на нижнем участке. Для этого переставляем плечи у двух развилок: «Жаркое как подошва?» и «Другая порция еще хуже?».

В результате шести рокировок наша мечта сбылась! Неэргономичная схема на рис. 75 превратилась в хорошую (эргономичную) схему на рис. 76.

Подведем итоги. Выпрямляя главный маршрут, мы делаем алгоритм более наглядным и легким для понимания. Главный маршрут — путеводная нить алгоритма, позволяющая быстрее уяснить суть дела и не заблудиться в хороводе развилок.



**Рис. 75.** Плохой дракон-алгоритм. Главный маршрут (жирная линия) все время петляет и делает зигзаги. Его трудно проследить взглядом



**Рис. 76.** Хороший дракон-алгоритм. Он получен в результате улучшения схемы на рис. 75. Главный маршрут прямой как стрела. Выполняется правило «Чем правее, тем хуже».

А теперь – самое главное. Мы осуществили выпрямление главного маршрута не случайно, не по принципу «Что хочу, то и ворочу!», а на основании строгого математического закона — закона рокировки. Напомним суть закона: рокировка — равносильное преобразование алгоритма. При рокировке смысл алгоритма не меняется.

Полученный результат чрезвычайно важен. Мы воочию убедились, что рокировка позволяет улучшить наглядность и эргономичность алгоритмов.

Закон рокировки придает нашим эргономическим действиям (позволяющим выпрямить «кривой» главный маршрут) математическую строгость и точность

## ПОПУТНЫЙ СОВЕТ РАЗРАБОТЧИКУ МЕДИЦИНСКИХ АЛГОРИТМОВ

В медицинской литературе процедурные и декларативные знания не разграничены и перемешаны, что вносит серьезные трудности при разработке алгоритмов.

В медицинских текстах и публикациях завершенные алгоритмы в большинстве случаев (такова традиция) отсутствуют. Обычно они представлены не целиком, а виде мелких и мельчайших фрагментов (вкраплений), которые переплетаются с другой информацией.

1. Чтобы вычленить алгоритмические знания (вкрапления), их необходимо очистить, отсортировать и отделить от других тем и материалов.
2. Чтобы извлечь медицинский алгоритм из медицинских публикаций, необходимо:
  - Отделить процедурный текст от декларативного.
  - Выявить все условия, в том числе пропущенные в тексте.
  - Выделить все сложные условия, раздробить и заменить каждое из них на несколько простых условий.
  - Обеспечить эргономическую упорядоченность алгоритма, проверив строгое соблюдение картографического принципа

Картографический принцип обеспечивает порядок, эргономическую наглядность и концептуальное единство во всех частях медицинского алгоритма.

## ВЫВОДЫ

1. Язык ДРАКОН позволяет существенно упростить сложную логику медицинских алгоритмов и представить ее в виде наглядных зрительных образов.
2. Вместо утомительной работы с логическими формулами врачу достаточно запомнить два мнемонических понятия (Мачта и Лестница) и соответствующие им зрительные образы.
3. Графические логические схемы подчиняются следующим закономерностям:
  - Стандартная схема ИЛИ (лестница) служит для представления двух и более позитивных вопросов. Шампур помечают словом Да.
  - Нестандартная схема ИЛИ (мачта) предназначена для изображения двух и более негативных вопросов. Шампур маркируют словом Нет.

- Стандартную схему И (мачту) используют, чтобы показать два и более позитивных вопроса. Возле шампуром пишут Да.
  - Нестандартную схему И (лестницу) применяют, чтобы изобразить два и более негативных вопроса. Рядом с шампуром пишут Нет.
4. Графическая схема логического отрицания позволяет изменять и варьировать логические условия в схемах ИЛИ и И, обогащая выразительные возможности графической логики.
  5. Рокировка — равносильное преобразование медицинского алгоритма, которое видоизменяет внешний графический облик алгоритма, не меняя его сути. При рокировке смысл алгоритма не меняется.
  6. Рокировка позволяет улучшить наглядность и эргономичность медицинских алгоритмов и устраниТЬ нарушение картографического принципа.
  7. Закон рокировки придает эргономическим действиям разработчика алгоритма (направленным на соблюдение картографического принципа) математическую строгость и точность.
  8. В языке ДРАКОН реализован принцип невидимой математики. Это значит, что логические операции дизъюнкции  $\vee$ , конъюнкции  $\&$ , отрицания  $\neg$  и законы Августа де Моргана становятся полностью «невидимыми» для врача и реализуются в медицинских алгоритмах с помощью интуитивно понятной графики, что значительно облегчает работу медицинского персонала.

## Глава 9

# ПОВТОРЕНИЕ МЕДИЦИНСКИХ ДЕЙСТВИЙ, ИЛИ ЦИКЛ

### ЧТО ТАКОЕ ЦИКЛ

В медицине часто приходится повторять одни и те же действия. Как изобразить такое повторение? Для этого используют специальный прием, который называется **цикл**.

Циклические алгоритмы — вещь довольно сложная. Чтобы не затруднять читателя, поясним идею на шутливом сказочном примере. Как говорил великий Блез Паскаль, «предмет математики настолько серьезен, что полезно не упускать случая сделать его немного занимательным» [140].

### РАССКАЗ О ЗМЕЕ ГОРЫНЫЧЕ

Предположим, у Змея Горыныча три головы. Чтобы победить Змея, Илья Муромец должен отсечь злодею все головы. То есть три раза исполнить команду «Отруби голову» (рис. 77). Если у Змея пять голов, эту команду придется повторить пять раз (рис. 78).

А если у Змея сто или тысяча голов? Тогда — если действовать в лоб — придется написать подряд сто или тысячу команд. Ясно, что это нелепый путь. Чтобы изложить алгоритм подобным образом, никакой бумаги не хватит!

Чтобы не писать слишком много одинаковых команд, нужно использовать цикл. Цикл — это повторяющееся выполнение одних и тех же команд. Фокус в том, что команда в алгоритме записывается *один* раз, а исполняется *много* раз — столько, сколько нужно.

# Змей Горыныч и циклические алгоритмы

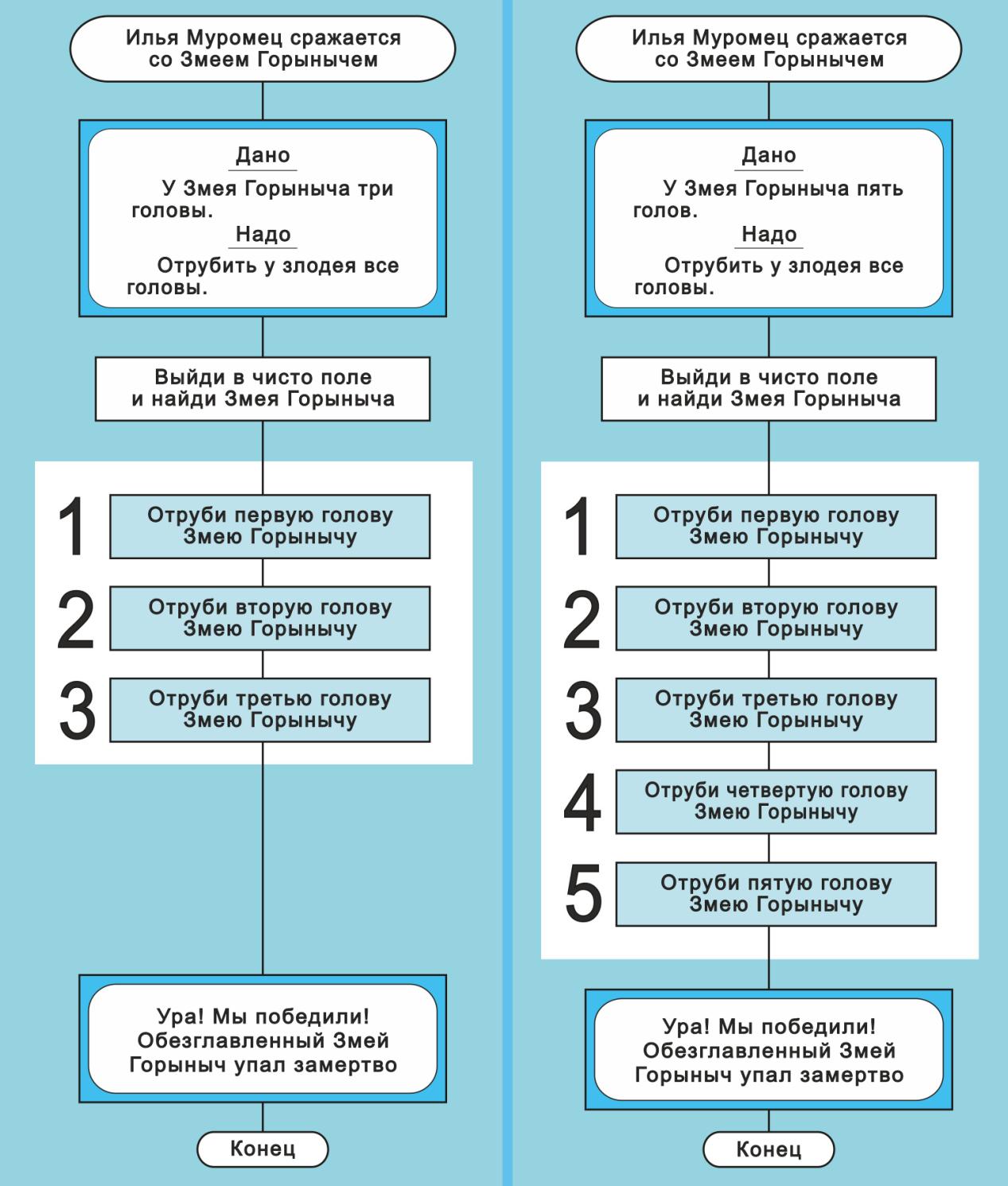


Рис. 77. Плохой (очень длинный) алгоритм. Чтобы отрубить три головы, приходится писать три команды «Отруби голову».

Рис. 78. Плохой (очень длинный) алгоритм. Чтобы отрубить пять голов, надо написать пять команд «Отруби голову». А если у Змея сто голов?

# Змей Горыныч и циклические алгоритмы



Это и есть ЦИКЛ, которого боится Змей Горыныч

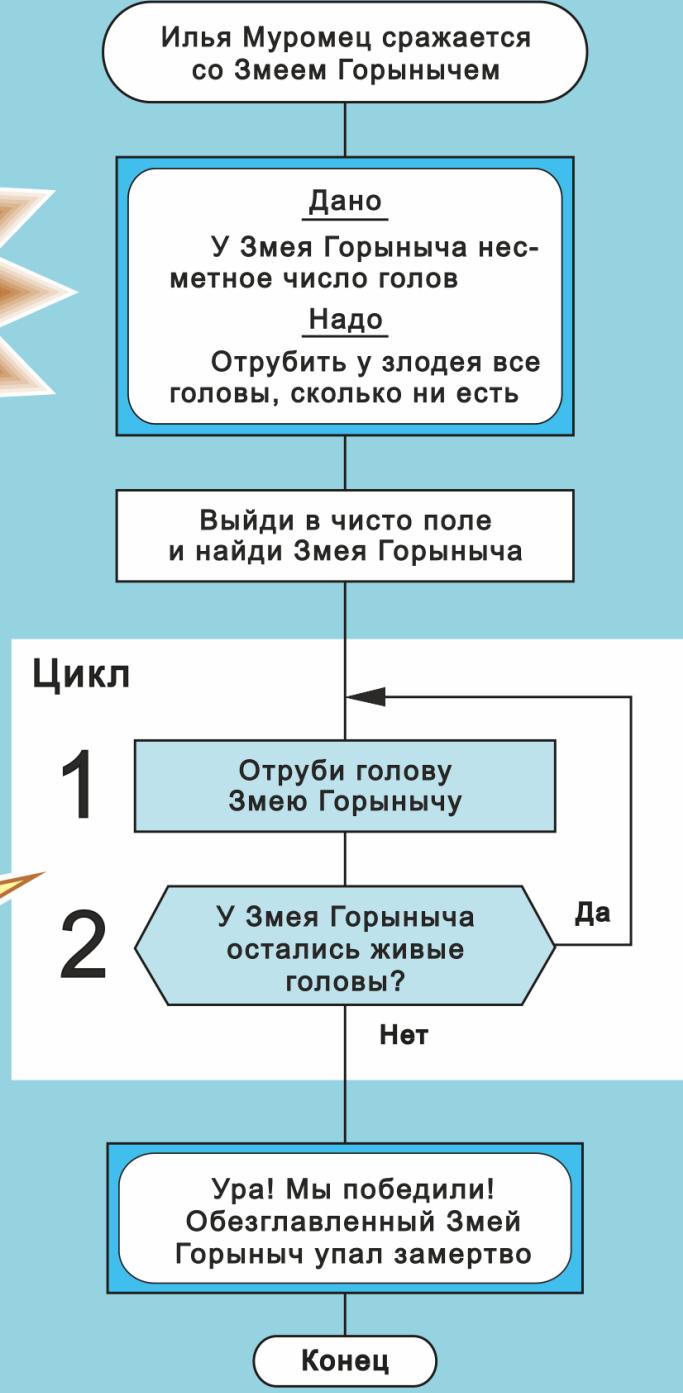


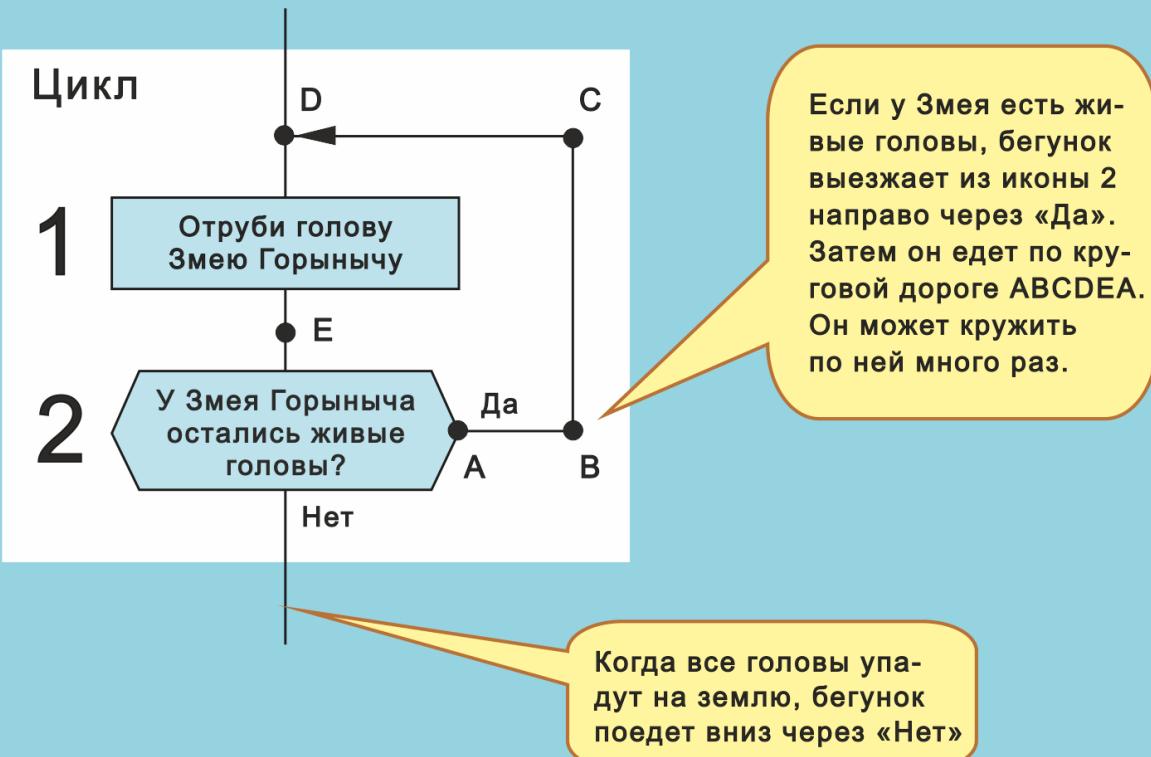
Рис. 79. Хороший (короткий) алгоритм.

А почему он короткий? Потому что в нем есть ЦИКЛ.

Циклический алгоритм позволяет отрубить любое число голов.

При этом нужна всего одна команда «Отруби голову».

# Змей Горыныч и циклические алгоритмы



## УСЛОВИЕ ПРОДОЛЖЕНИЯ ЦИКЛА

У Змея Горыныча  
остались живые головы?

=

Да

## УСЛОВИЕ ОКОНЧАНИЯ ЦИКЛА

У Змея Горыныча  
остались живые головы?

=

Нет

**Рис. 80.** Цикл — это замечательно!  
Команды 1 и 2 повторяются много раз,  
если выполняется УСЛОВИЕ ПРОДОЛЖЕНИЯ ЦИКЛА.

Как работает цикл? Предположим, у Змея только одна голова, которую мы победили с помощью команды 1 (рис. 79). В этом случае на вопрос 2: «У Змея Горыныча остались живые головы?» — отвечаем Нет, и алгоритм заканчивается.

Если же у Змея есть уцелевшие головы, отвечаем Да и, проходя через стрелку, снова попадаем на вход иконы 1. После чего рубим следующую голову. Еще раз задаем вопрос 2. Если у Змея по-прежнему остались головы, отвечаем Да и вновь возвращаемся на вход иконы 1.

Таким образом, исполнение команд 1 и 2 может продолжаться и сто, и двести, и тысячу раз — до тех пор, пока наш алгоритм (управляющий Ильей Муромцем) не отсчет у Змея все головы до последней — см. также рис. 80.

## УСЛОВИЕ ПРОДОЛЖЕНИЯ И ОКОНЧАНИЯ ЦИКЛА

Следует различать два понятия (рис. 80):

- условие продолжения цикла,
- условие окончания цикла.

Пока выполняется первое условие, действие повторяется снова и снова. В нашем примере это значит, что Илья Муромец безостановочно рубит головы.

Как только исчезает первое условие, сразу же возникает условие окончания цикла:

У Змея Горыныча остались живые головы? = Нет

Рубить больше нечего. Бегунок на рис. 80 перестает крутиться по круговой дороге и выскользывает из иконы Вопрос вниз через Нет.

## КАК ИЗОБРАЗИТЬ ПОВТОРЕНИЕ ДЕЙСТВИЙ В МЕДИЦИНЕ

При реанимации иногда применяют вдохи «рот в рот». Чтобы выполнить вдох пострадавшему, спасатель должен произвести энергичный выдох.

Предположим, нужно сделать пострадавшему не один искусственный вдох, а много. Как это показать на чертеже? Для этого используют графический цикл.

На рис. 81 в нижней иконе Действие сказано: «Выполнни вдох пострадавшему продолжительностью 1 сек.» Выполняя вдох, нужно следить за грудной клеткой пациента. Если она неподвижна (не поднимается), искусственный вдох следует повторять снова и снова. Это легко проследить по алгоритму. Если грудная клетка не шевелится, мы выходим из иконы Вопрос через Нет. А что дальше?

Дальше поворачиваем наверх и по стрелке вновь попадаем в икону «Выполнни вдох...». В результате вдох повторяется опять и опять — до тех пор, пока грудная клетка пострадавшего не начнет ритмично подыматься и опускаться.

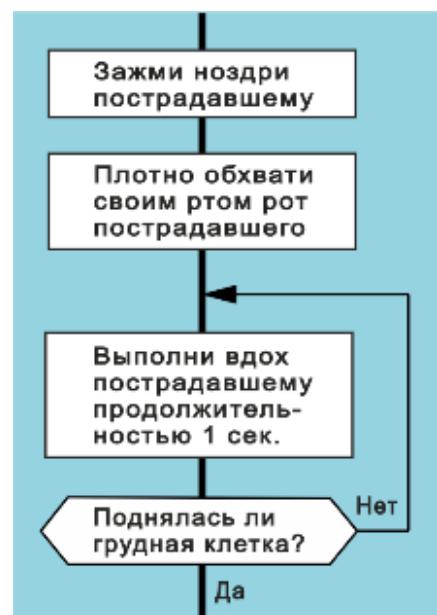


Рис. 81. Применение цикла при реанимации [198]

Таким образом, термин «цикл» используется в медицинских алгоритмах, чтобы обозначить циклическое повторение врачебных действий. На рис. 21 (пункт 3) представлена макроикона «Цикл».

Что такое  
цикл

- Это повторение одного и того же действия (действий). Выигрыш в том, что действия в алгоритме записываются один раз, а выполняются много раз.
- Повторение действий прекращается, когда будет выполнено условие, указанное в иконе Вопрос.

## ВЫВОДЫ

1. Если нужно несколько раз повторять медицинскую процедуру, следует использовать графическую фигуру «цикл» (см. макроикону на рис. 21, п. 3).
2. Стрелка цикла всегда закручивается против часовой стрелки.
3. При анализе повторяющихся действий необходимо исключить ошибки и путаницу при использовании понятий:
  - условие продолжения цикла,
  - условие окончания цикла

# Глава 10

## СОВМЕСТНАЯ РАБОТА ВРАЧЕЙ

### РАБОТА ГРУППЫ ВРАЧЕЙ

До сих пор мы рассматривали одиночную работу врача. Однако в сложных случаях врачи объединяются в бригады и осуществляют медицинскую помощь совместными усилиями.

Отличие в том, что каждый член бригады выполняет свою часть работы не изолированно, а согласованно с другими медиками. Необходима слаженная, синхронная и хорошо скоординированная деятельность персонала. Члены группы работают одновременно и параллельно.

Параллельную работу медицинских работников нужно уметь изображать в виде алгоритма. Подобные алгоритмы называются *параллельными*.

В данной главе представлены параллельные алгоритмы для двух врачей.

### СОВМЕСТНАЯ РАБОТА БРИГАДЫ СКОРОЙ ПОМОЩИ

На рис. 82 показана деятельность двух работников скорой помощи. Алгоритм определяет порядок выполнения неотложных действий по спасению пострадавшего мотоциклиста. Он находится без сознания после дорожной аварии с подозрением на перелом позвоночника.

Помощь мотоциклиstu оказывают два спасателя. Они работают одновременно и параллельно. На рисунке им присвоены номера 1 и 2. Действия первого спасателя описаны в левом столбце, второго — в правом.

Важно отметить, что действия спасателей скоординированы во времени. Координация показана с помощью двух икон:

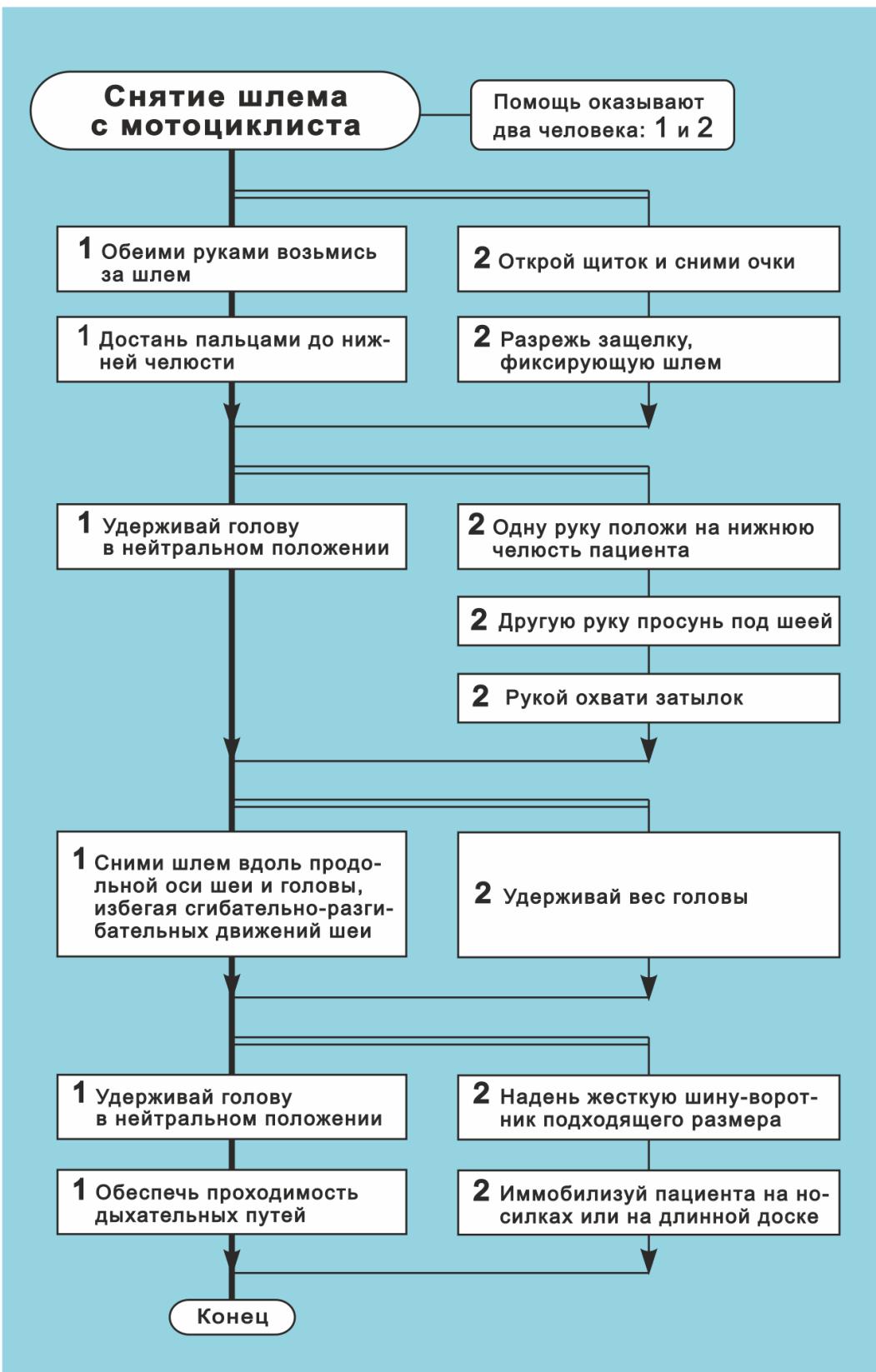


Рис. 82. Алгоритм, описывающий работу двух работников скорой помощи [141]

- Начало синхронной работы спасателей обозначается двойной линией.
- Конец синхронного участка изображают одиночной (горизонтальной) линией, в которую сверху вонзаются две стрелки.

## ПРАВИЛА НУМЕРАЦИИ СПЕЦИАЛИСТОВ

При разработке параллельного медицинского алгоритма необходимо заранее установить четкую нумерацию членов врачебной бригады. Например, для бригады из двух специалистов, необходимо:

- Справа от иконы Заголовок присоединить икону Пояснение с надписью: «Помощь оказывают два человека: **1 и 2**».
- В левой части всех без исключения икон алгоритма поставить номер ответственного за данную операцию члена бригады: **1 или 2**.
- Цифры должны быть крупными и жирными (в пределах разумного), чтобы они сразу бросались в глаза и не вызывали затруднений у читателей.
- Если какую-то операцию медики выполняют вдвоем, в соответствующей иконе слева пишут два номера через запятую: **1, 2**.

Соответствующие иллюстрации показаны на рис. 82 и 83.

*Примечание.* Икона Пояснение приведена в справочнике на рис. 20, пункт 19.

## ДВУХПОТОЧНЫЙ УЧАСТОК

*Двухпоточный участок* — часть параллельного алгоритма, обладающая следующими свойствами:

- Начало участка обозначено двойной линией, а конец — одиночной.
- В начале участка единый маршрут (поток) раздваивается и превращается в два: левый и правый. В конце участка, наоборот, два потока объединяются и превращаются в один.
- Двухпоточный участок решает строго определенную функциональную задачу, требующую синхронных действий членов медицинской бригады.

Сверхзадача алгоритма на рис. 82 соответствует принципу «Не навреди». Любое небрежное или неосторожное действие спасателей может усугубить состояние мотоциклиста и нанести ему непоправимый вред. Исходя из сверхзадачи, алгоритм разбит на четыре двухпоточных участка, каждый из которых выполняет свою частную задачу.

- **1-й участок.** Выполнить подготовительные операции: открыть щиток, снять очки, разрезать защелку.
- **2-й участок.** Установить руки второго спасателя в нужное положение, чтобы подготовиться 3-му участку (где надо удерживать вес головы).
- **3-й участок.** Снять шлем мотоциклиста.
- **4-й участок.** Надеть жесткую шину-воротник и иммобилизовать пострадавшего на носилках или доске.

## СОВМЕСТНАЯ РАБОТА ПРИ ОТСАСЫВАНИИ МЕКОНИЯ ИЗ ТРАХЕИ НОВОРОЖДЕННОГО

Рассмотрим более сложный случай. В алгоритме на рис. 83 применяются три интересных приема:

- совместная работа врачей,
- цикл,
- икона Вставка.

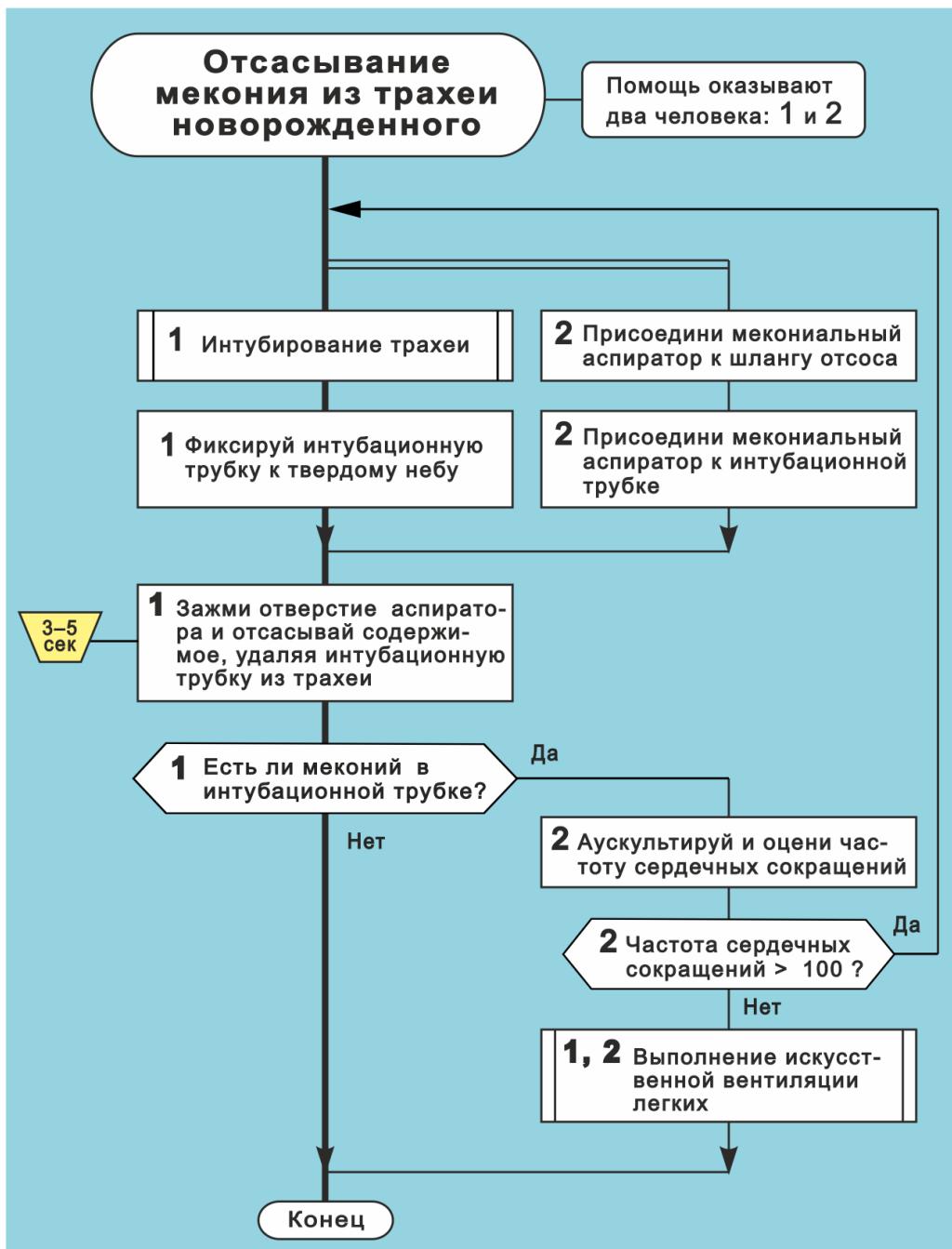


Рис. 83. Алгоритм «Отсасывание мекония из трахеи новорожденного» [142]

Алгоритм выполняют два врача, которым на чертеже присвоены номера 1 и 2. Действия первого врача описаны в левом столбце, второго — в правом.

Действия врачей согласованы во времени. Начало совместной работы показано в виде двойной линии, конец — одиночной.

Тонкость состоит в том, что бегунок в верхней части чертежа (ниже двойной линии) раздваивается и одновременно проходит по двум маршрутам: левому и правому. Затем (в конце двухпоточного участка) два бегунка сливаются в один. Это происходит выше иконы «Зажми отверстие аспиратора...». Далее следует один-единственный бегунок.

Тем самым бегунок наглядно демонстрирует групповой характер работы врачей.

Обратите внимание. В данном алгоритме совместная работа отражается двумя способами, именно:

- двухпоточный участок.

### ● нумерация специалистов,

В верхней части рисунка используются оба способа, а в нижней только один — нумерация специалистов.

Перейдем к анализу цикла. Как его обнаружить?

Найдите горизонтальную стрелку (именно горизонтальную). Затем пройдите по длинной линии, исходящей из иконы Вопрос «Частота сердечных сокращений > 100?» через Да.

Проследите карандашом путь, идущий после стрелки вниз через иконы:

- «Интурирование трахеи».
- «Фиксируй интубационную трубку...».
- «Зажми отверстие аспиратора...».
- «Есть ли меконий в интубационной трубке?» Да.
- «Аускультируй и оцени частоту...».
- «Частота сердечных сокращений > 100?» Да.

Если на оба вопроса постоянно дается ответ Да, ваш карандаш будет двигаться по кругу против часовой стрелки, описывая все новые и новые круги и никогда не остановится. Получится так называемый бесконечный цикл, из которого нет выхода.

В реальности такого быть не может по двум причинам.

В благополучном случае весь меконий из трахеи будет отсосан, и мы выйдем из иконы Вопрос «Есть ли меконий в интубационной трубке?» через Да.

В неблагоприятном случае (когда частота сердечных сокращений новорожденного менее 100 ударов в минуту, то есть новорожденный неактивный), следует незамедлительно начать искусственную вентиляцию легких. На алгоритме хорошо видно, что из иконы Вопрос «Частота сердечных сокращений > 100?» мы выходим через Да.

**Вставка.** На рис. 83 используются две иконы Вставка:

- «Интурирование трахеи»,
- «Выполнение искусственной вентиляции легких».

Каждая из них представляет собой сложный алгоритм, который обязательно должен быть раскрыт где-нибудь в подходящем для этого месте. Причем читатель должен иметь возможность попасть в это место и ознакомиться с алгоритмом. Если такой возможности нет, значит, Вставка превращается в шараду: «Пойди туда, не знаю куда».

В таких случаях говорят, что медицинский алгоритм выполнен некачественно и раскрыт не полностью. Это грубая алгоритмическая ошибка.

**Примечание.** Описание иконы Вставка дано также в главе 7 на рис. 23 и 24.

## ИКОНА «ВРЕМЯ»

Давайте посмотрим на рисунок 83 через увеличительное стекло и найдем желтую икону «Время». Чтобы она всегда была под рукой, перенесем ее вместе с соседкой на рис. 84.

Мы видим, что икона Время всегда образует пару. Она прицеплена слева к своей хозяйке — иконе Действие и указывает **длительность выполнения операции**.

Но это не все. Икона Время выполняет более тонкую и важную функцию. Поскольку речь идет о спасении жизни новорожденного,

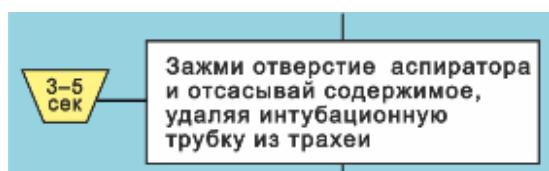


Рис. 84. Икона Время

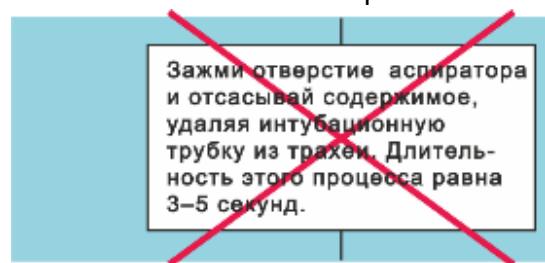


Рис. 85. Вариант без иконы Время

она подсказывает врачу, что он обязан выполнить действие БЫСТРО, за 3–5 секунд, потому что при реанимации счет идет на секунды.

Конечно, можно поступить по-другому и обойтись без иконы Время. Для этого нужно:

- перетащить информацию о длительности в «хозяйскую» икону Действие;
- в последней написать: «Зажми отверстие аспиратора и отсасывай содержимое, удаляя интубационную трубку из трахеи. Длительность этого процесса равна 3–5 секунд». Этот вариант показан на рис. 85.

Однако такой способ намного хуже. Беда в том, что высказывание получилось слишком длинное и неудобное для чтения. Чтобы облегчить труд читателя, длинную словесную глыбу лучше разбить на две части. И поместить их в две разные иконы — Время и Действие. Именно так сделано на рис. 84.

При этом необходимость читать длинный и громоздкий текст отпадает. Икона Время быстро и элегантно предоставляет врачу важную информацию, избавляя его от необходимости выискивать предупреждение о времени в темных закоулках мудреного словесного абзаца.

Таким образом, икона Время позволяет качественно улучшить алгоритм. Она создает удобное средство для синхронизации и координации временных соотношений в параллельных медицинских алгоритмах.

Чтобы медицинская наука интенсивно развивалась, она должна овладеть новым языком для описания медицинских знаний. Этот язык должен быть очень хорошим. Он должен позволять создавать высококачественные медицинские алгоритмы — доступные, удобные, легкие для понимания. Язык ДРАКОН призван способствовать достижению этой цели. На отдельных частных примерах мы стремимся показать различные возможности языка.

## ВЫВОДЫ

1. В сложных случаях врачи объединяются в бригады и осуществляют медицинскую помощь коллективно.
2. Параллельные медицинские алгоритмы служат для описания синхронной, точно скоординированной работы членов медицинской бригады.
3. При разработке параллельного медицинского алгоритма используют:
  - правила нумерации специалистов,
  - правила двухпоточного участка.
4. Начало двухпоточного участка обозначают двойной линией, конец — одиночной.
5. Параллельный алгоритм может содержать несколько синхронных (двухпоточных) участков, соединенных последовательно.
6. Координацию длительности операций удобно задавать с помощью иконы Время.

# Глава 11

## НОВЫЙ СИЛУЭТ МЕДИЦИНСКОГО АЛГОРИТМА

### ПРИМИТИВ И СИЛУЭТ

Язык ДРАКОН имеет две алгоритмические конструкции:

- *примитив* (для описания простых медицинских алгоритмов),
- *силуэт* (для сложных и сверхсложных).

В предыдущих главах мы рассматривали простейшие случаи и использовали конструкцию «примитив». Примеры примитивов представлены на рис. 1-4, 23-26, 28, 75-79, 82, 83.

На рис. 8, 9, 14, 15, 29-34, 36-75, 80, 81, 84, 85 показаны фрагменты алгоритмов, которые можно применять где угодно — и в примитиве, и в силуэте.

Чем отличается примитив от фрагмента? Примитив — простой, но полностью законченный алгоритм, имеющий иконы Заголовок и Конец. Фрагмент — недоделанная часть алгоритма, в ней нет ни Заголовка, ни Конца.

### ПРИНЦИПИАЛЬНЫЙ НЕДОСТАТОК ПРИМИТИВА

Важно помнить, что примитив имеет ограниченные возможности. Во многих случаях (за исключением самых простых) он не позволяет изобразить ЛЮБОЙ медицинский алгоритм.

От этого недочета свободен силуэт, который можно применять во всех без исключения случаях, в том числе в наиболее сложных и изощренных. Однако для самых простых случаев все-таки выгоднее использовать примитив.

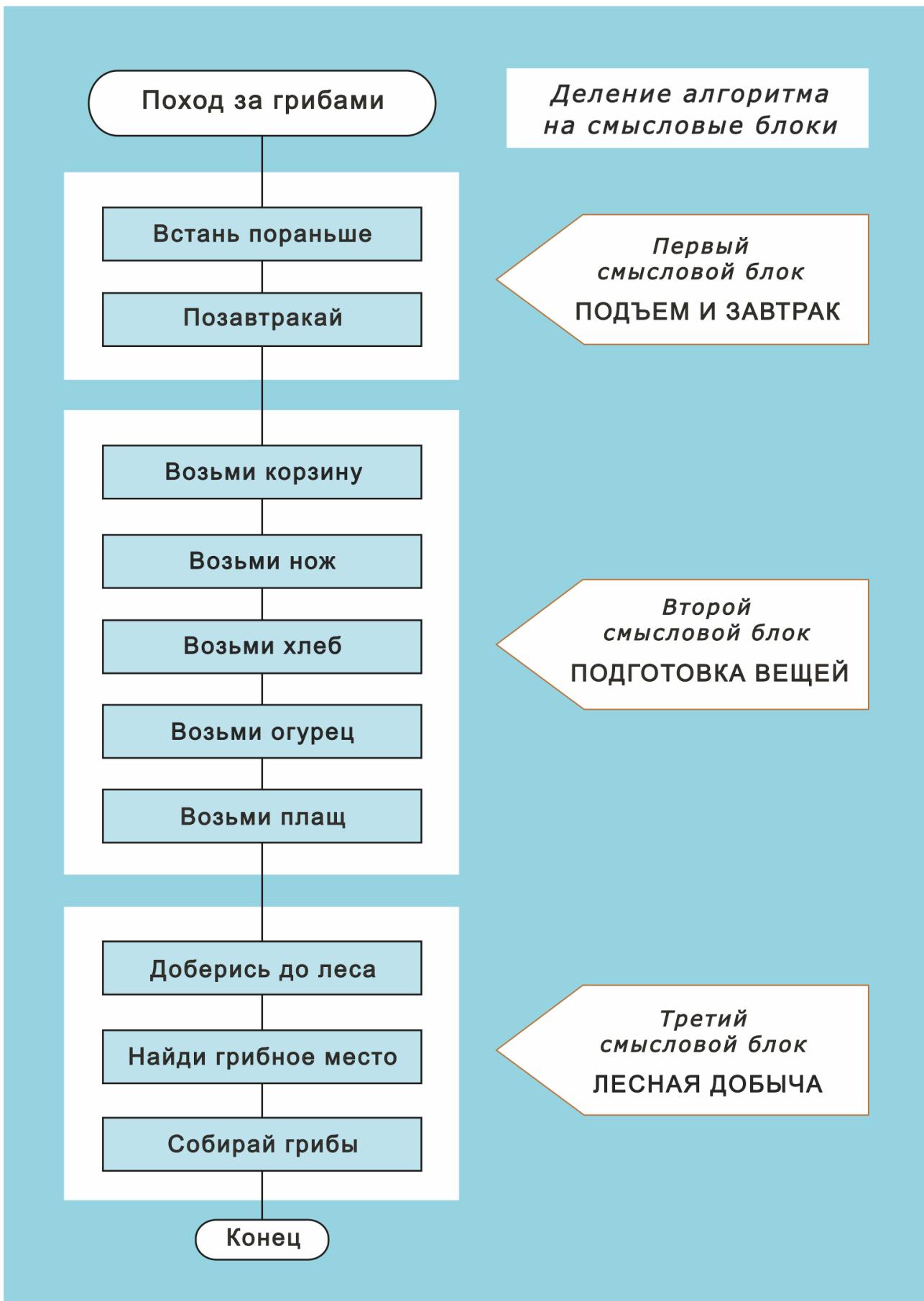


Рис. 86. Алгоритм «Поход за грибами»

## ЧТО ТАКОЕ СИЛУЭТ

Силуэт — чрезвычайно мощная алгоритмическая конструкция, обладающая большими выразительными возможностями. Она позволяет описывать медицинские алгоритмы любой сложности, причем делать это в привлекательной для врачей форме.

Силуэт является основным достоинством языка ДРАКОН, его главным оружием.

Основным элементом силуэта является «ветка». С нее-то мы и начнем.

## ВЕТКА

На рис. 86 представлен алгоритм «Поход за грибами», содержащий довольно большую последовательность действий. Иной раз такая последовательность может оказаться чрезмерно длинной и утомительной для чтения.

Можно ли облегчить восприятие и анализ подобных «длиннujących» алгоритмов? Можно ли сделать «долговязый» алгоритм обозримым и удобным для быстрого понимания?

Да, можно. Чтобы облегчить работу читателя и сделать алгоритм эргономичным, надо заблаговременно разрезать «длинную кишку» и разбить ее на смысловые части. Сделать это нетрудно. «Поход за грибами» (рис. 86) — это алгоритмический рассказ, в котором можно выделить три крупных куска, три самостоятельных темы:

- Подъем и завтрак.
- Подготовка вещей.
- Лесная добыча.

Каждую тему можно нарисовать в виде ветки. Результат изображен на рис. 87.

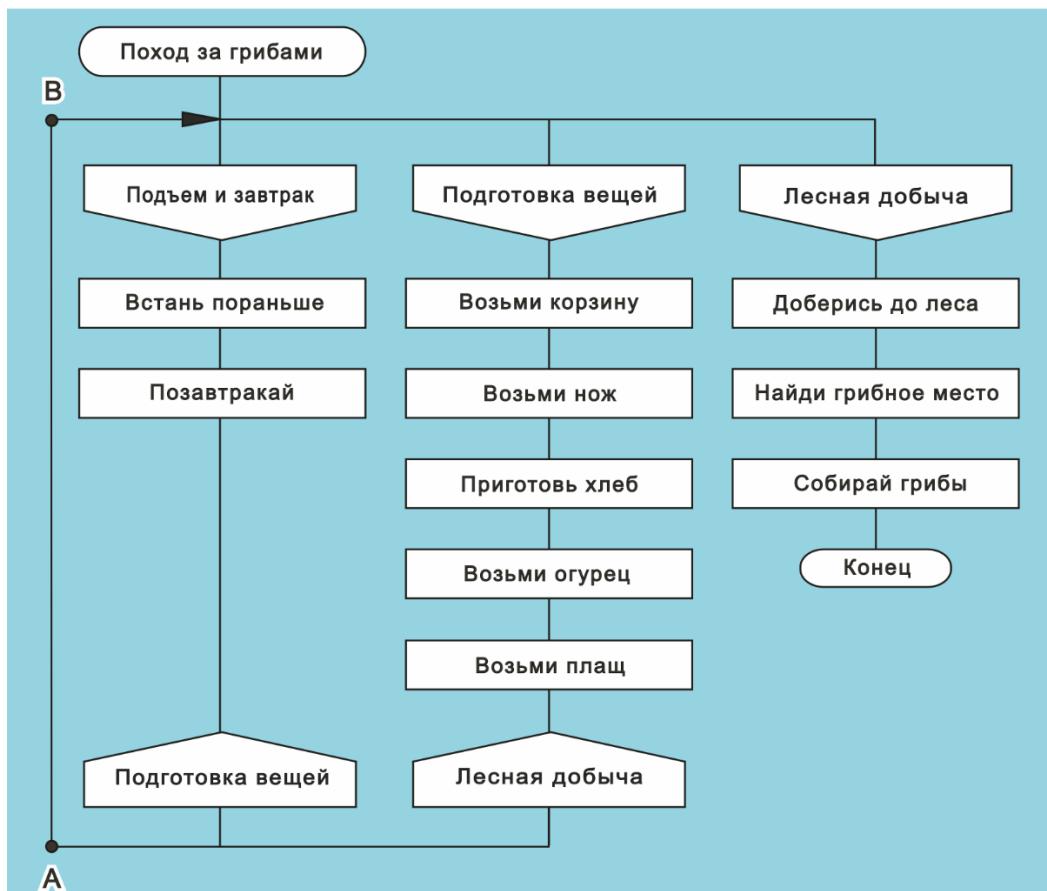


Рис. 87. Алгоритм силуэт «Поход за грибами». Сравни с рис. 86

Названия двух новых икон приведены в таблице.

Икона	Название иконы	Пояснение
	Имя ветки	Обозначает начало ветки
	Адрес	Обозначает конец любой ветки, кроме последней

*Вопрос.* Где начало и конец у первой ветки на рис. 87?

*Ответ.* Начало – икона «Подъем и завтрак». Конец – «Подготовка вещей». Между началом и концом размещается тело ветки. Оно содержит команды «Встань пораньше» и «Позавтракай».

Зачем  
нужна ветка

Чтобы помочь врачу разбить медицинский алгоритм на смысловые части. И, что очень важно, дать частям удобные и точные названия. Название каждой части пишут в иконе «Имя ветки».

Чтобы разделить проблему на несколько смысловых частей (например, на пять), разбейте алгоритм на пять веток.

Что такое  
ветка

Это смысловая часть алгоритма, которая содержит:

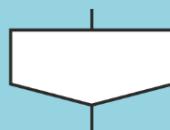
- Икону «Имя ветки» (в ней пишут название смысловой части).
- Тело ветки, состоящее из команд, записанных в иконах.
- Одну или несколько икон Адрес (в любой ветке, кроме последней).
- Икону Конец (в последней ветке).

## ШПАРГАЛКА

Икона «Имя ветки» очень похожа на икону «Вариант». С непривычки их можно перепутать. Чтобы избежать досадных ошибок, сравните иконы на рис. 88 и 89.

### Чем отличается икона «Имя ветки» от иконы «Вариант»?

Икона «Имя ветки»



Икона «Вариант»

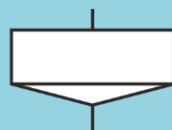


Рис. 88. У иконы «Имя ветки» одна горизонтальная линия

Рис. 89. У иконы «Вариант» две горизонтальные линии

## КАК ЧИТАТЬ СИЛУЭТ?

Сначала читают Заголовок, затем по шампуру — крайнюю левую ветку. В самом низу находится икона «Адрес». Зачем она нужна?

В ней пишут Имя следующей по порядку исполнения ветки. Икона Адрес не позволит вам сбиться с пути. Она подсказывает, какую ветку следует читать дальше.

Вход в ветку возможен только через ее начало. Выход из последней ветки осуществляется через икону Конец.

## ШАПКА

Многие алгоритмы очень сложны. Чтобы разобраться в тонкостях такого алгоритма, нужно прилагать большие усилия и тратить много времени.

Подобную практику следует признать порочной. Алгоритмы можно и нужно сделать легкими для восприятия (эргономичными). Для этого надо давать читателю маленькие, но умные подсказки, проглотив которые, он мог бы легко сориентироваться в задаче и быстро понять, что к чему. Одной из таких подсказок служит *шапка*. Название объясняется тем, что шапка находится вверху, «на голове» у алгоритма.

Назначение шапки — помочь читателю мгновенно (за несколько секунд) сориентироваться в задаче и, расчленив ее на части, увидеть структуру алгоритма (рис. 90). Причем увидеть не в фигуральном смысле слова, не с помощью воображения, не духовным оком, а своими двумя глазами — на бумаге или экране.



Рис. 90. Шапка — важный элемент силуэта. Она показывает структуру силуэта

Что такое  
шапка?

Это верхняя часть силуэта, которая включает Заголовок алгоритма и комплект икон «Имя ветки».

## ТРИ «ЦАРСКИХ» ВОПРОСА

Сталкиваясь с новой незнакомой задачей, мы обычно желаем получить ответ на три царских (наиболее важных) вопроса:

1. Как называется задача?
2. Из скольких частей она состоит?
3. Как называется каждая часть?

Именно с этих вопросов начинается наше знакомство с задачей при рациональном подходе к делу.

Эргономическая хитрость состоит в том, что шапка, угадывая тайное желание читателя, дает ему подсказку — ответ на все царские вопросы.

Вот ответы для рис. 87.

- Как называется задача? (*Читаем Заголовок алгоритма*).  
Поход за грибами.
- Из скольких частей она состоит? (*Считаем иконы «Имя ветки»*).  
Из трех.
- Как называется каждая часть? (*Читаем текст в иконах «Имя ветки»*).

1. Подъем и завтрак.
2. Подготовка вещей.
3. Лесная добыча.

## КАК БЕГУНОК ДВИЖЕТСЯ ПО СИЛУЭТУ

Вспомним, что бегунок — воображаемая точка, которая поочередно пробегает все иконы одного из маршрутов, перемещаясь из начала в конец. Как он ведет себя в силуэте?

Как обычно — мчится по алгоритмической дорожке от иконы Заголовок до иконы Конец. Выехав из Заголовка бегунок скользит вниз по крайней левой ветке. Он движется через станции (рис. 87):

- Подъем и завтрак.
- Встань пораньше.
- Позавтракай.
- Подготовка вещей.

Икона Адрес — последняя станция первой ветки. Куда ехать дальше? Ответ содержится внутри самой иконы. Эта икона потому и зовется «Адрес», что сообщает адрес (название) следующей станции. В данном случае она говорит: следующая станция называется «Подготовка вещей».

Из рис. 87 видно, что данная станция находится в начале второй ветки. Но как бегунок доберется туда? По какой линии?

Ответ прост. Выехав из иконы Адрес, бегунок сворачивает налево и попадает в точку *A* (рис. 87). Потом движется вверх к точке *B*. Затем едет направо по стрелке и въезжает в верхнюю икону «Подготовка вещей».

Дальше все происходит аналогично. Бегунок скользит вниз по второй ветке. Добравшись до иконы «Адрес», узнает адрес следующей станции («Лесная добыча»). Затем огибает схему по линии *AB*, попадает в начало третьей ветки и спускается по ней до конца. На этом выполнение алгоритма заканчивается.

## В ЧЕМ СЕКРЕТ ИКОНЫ «АДРЕС»

Сейчас мы поступим, как чеховский злоумышленник — будем разбирать рельсы. Имеются в виду линии, окаймляющие алгоритм на рис. 87. Сотрем линию *AB*. Уберем также горизонтальные линии (шины), проходящие через точки *A* и *B*. Результат представлен на рис. 91.

Как будет работать силуэт после этих исправлений?

К нашему удивлению, отсутствие рельсов никак не сказывается на работе алгоритма. В этом легко убедиться. Выехав из иконы Заголовок, бегунок движется вниз по крайней левой ветке. Опустившись до конца ветки, бегунок попадает в икону Адрес.

Казалось бы, это тупик. Рельсы кончились, и дальше ехать некуда. Но это не так. Ведь в иконе Адрес записан адрес следующей станции («Подготовка вещей»). Зная адрес, бегунок прыгнет туда, куда нужно — в начало второй ветки. И поедет вниз. Добравшись до конца второй ветки, он совершил второй прыжок. И попадет в третью ветку. И так далее.

Таким образом, икона «Адрес *N*» — это команда «Прыгни в начало ветки *N*». Проще говоря, данная команда передает приказ: «Брось данную ветку и начни выполнять ветку *N*». (Выполнять ветку — значит исполнять записанные в ней команды).

Что такое  
икона «Адрес N»



Это команда, заставляющая бегунок прыгнуть  
в начало ветки N

Переходя от рис. 87 к рис. 91, мы для упрощения стёрли несколько линий. И сразу убедились, что для работы алгоритма они совершенно не нужны. Маршрут бегунка определяют не они, а указания, записанные в иконе «Адрес».

Тем не менее, указанные линии не следует удалять по эргономическим соображениям. Дело в том, что обрамляющие линии зрительно «склеивают» разрозненные куски алгоритма. Они превращают их в приятный для глаза целостный зрительно-смысловый образ. И наоборот, устранение скрепляющих линий приводит к тому, что схема зрительно рассыпается на части, что сбивает с толку читателя (рис. 91).



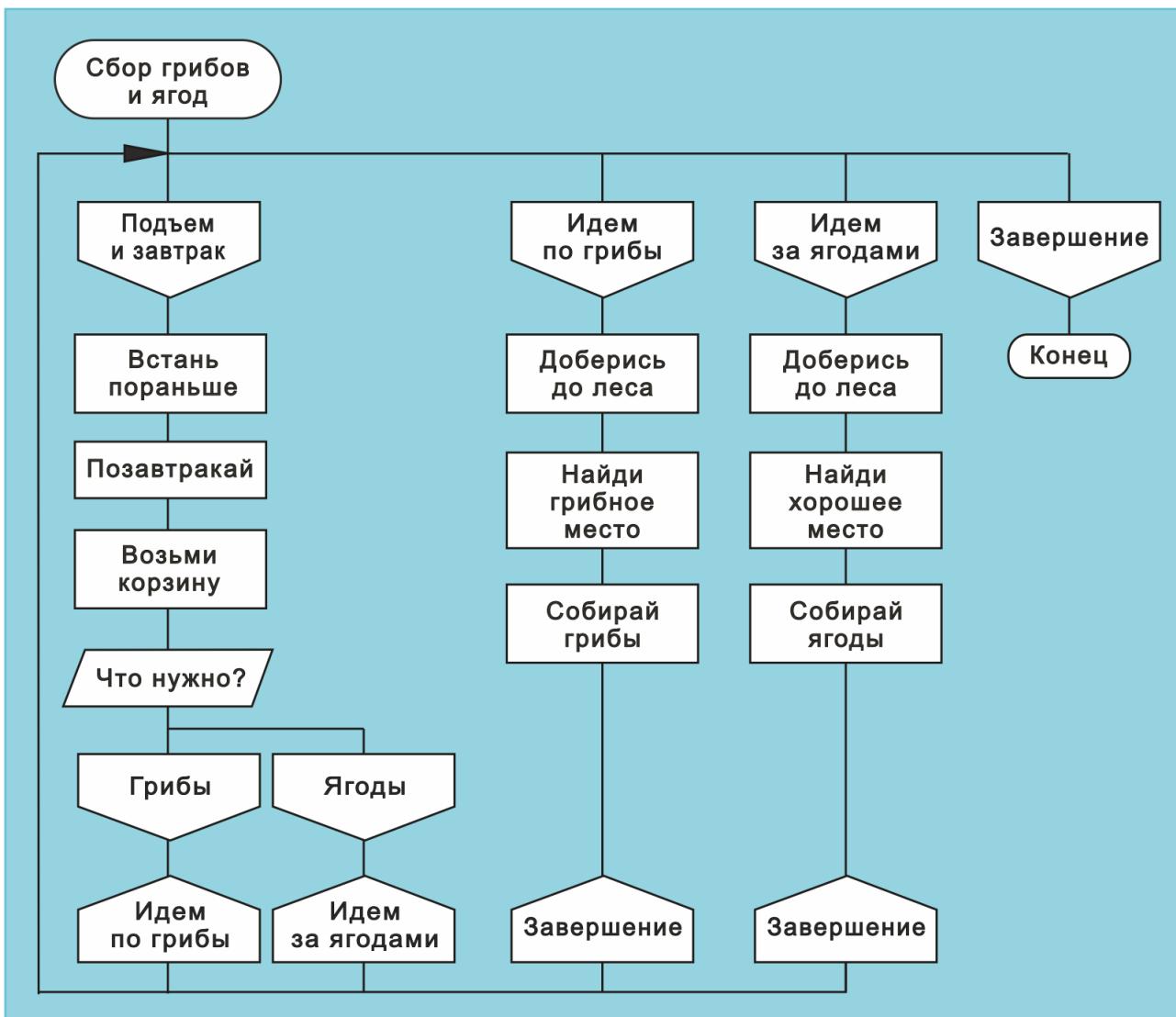
Рис. 91. Схема «с разобранными рельсами». Сравни с рис. 87

## ВХОД И ВЫХОДЫ ВЕТКИ

Ветка имеет один вход. И один или несколько выходов. На рис. 87 все ветки имеют только один выход. На рис. 92 ситуация иная — ветка «Подъем и завтрак» обладает двумя выходами:

- Идем по грибы.
- Идем за ягодами.

Это позволяет с помощью переключателя «Что нужно?» организовать разветвление и перейти в две различные ветки. Наличие у ветки нескольких выходов позволяет расширить выразительные возможности языка ДРАКОН.



**Рис. 92.** Ветка «Подъем и завтрак» имеет два выхода

## ПРАВИЛО ОДНОГО КОНЦА

Некоторые врачи по неопытности рисуют в алгоритме несколько концов. Это плохо.

Неприятность в том, что, глядя на схему, человеку трудно воспринимать алгоритм, имеющий несколько концов. Такой алгоритм похож на елку, увшанную «концами», как елочными игрушками. Подобная зрительная сцена распыляет внимание и мешает сосредоточиться на главном.

С эргономической точки зрения, желательно объединить все концы и превратить их в один. Это нетрудно сделать, как показано на рис. 92.

Чтобы объединить концы, в двух иконах Адрес написано слово «Завершение». Это слово означает прыжок в начало ветки «Завершение». Благодаря этому два «конца», уши которых торчат во второй и третьей ветках, объединяются и переносятся в последнюю ветку. Такой искусственный прием очень полезен — он позволяет соблюдать правило, запрещающее иметь в алгоритме несколько концов.

## КАК СЛЕДУЕТ РАСПОЛАГАТЬ ВЕТКИ НА ЧЕРТЕЖЕ

Ветки упорядочены двояко: логически и в пространстве. Логическая последовательность исполнения веток определяется подсказками, записанными в иконах «Адрес».

Однако логический порядок — это еще не все. Очень важно правильно расположить ветки в пространстве. Как это сделать?

Давайте мысленно перетасуем ветки, как колоду карт (рис. 91). И расположим их на чертеже в произвольном порядке. Легко сообразить, что подобное «перепутывание» веток никак не отразится на работе алгоритма. Ведь очередность работы веток зависит только от икон «Адрес». И совсем не зависит от расположения веток на листе бумаги. Словом, сколько ветки ни тасуй, получим тот же самый алгоритм.

Здесь есть тонкость. Перестановка веток не отражается на правильности алгоритма. Однако алгоритм должен быть не только правильным, но и понятным, эргономичным. Хаотичное расположение веток затрудняет понимание. А это недопустимо.

Поэтому нужно обязательно упорядочить ветки в пространстве чертежа. Удобнее всего расположить их слева направо в той последовательности, в какой они включаются в работу. Для этого служит правило: «Чем правее, тем позже». Оно означает: чем правее находится ветка, тем позже она работает.

## ЧТО МЫ УЗНАЛИ В ЭТОЙ ГЛАВЕ

Мы познакомились с силуэтом, наиболее важным, удобным и широко применяемым инструментом языка ДРАКОН.

В силуэте появились две новые иконы: «Имя ветки» и «Адрес». Таким образом, золотая десятка ДРАКОНа, с которой мы впервые повстречались на рис. 35, превратилась в золотую дюжину (рис. 93).

Преимущество в том, что один силуэт позволяет заменить несколько примитивов. Предположим, что алгоритм состоит из семи примитивов, логически связанных между собой. Семь примитивов можно легко превратить в семь веток, а последние объединить в единый силуэт.

Как превратить примитив в ветку? Очень просто.

Сначала надо заменить икону «Заголовок» примитива на икону «Имя ветки». А затем вместо иконы «Конец» примитива подставить икону «Адрес». Вот и все. Разумеется, превращение группы примитивов в один силуэт возможно лишь тогда, когда для этого имеются логические предпосылки.

В состав золотой дюжины входят десять икон (1-10) и две макроиконы (11 и 12).

Золотая дюжина  
языка ДРАКОН

1		Заголовок
2		Конец
3		Действие
4		Вопрос
5		Выбор
6		Вариант
7		Вставка
8		Комментарий
9		Имя ветки
10		Адрес
11		Развилка
12		Переключатель

Рис. 93. Двенадцать фигур, часто встречающихся в медицинских алгоритмах

## **ВЫВОДЫ**

1. Существуют два типа дракон-алгоритмов: примитив и силуэт.
2. Примитив предназначен для описания простых медицинских алгоритмов, силуэт — для сложных и сверхсложных.
3. Силуэт состоит из веток.
4. Ветка содержит две новые иконы (Имя ветки и Адрес), которые отсутствуют в примитиве.
5. Икона Имя ветки обозначает начало ветки.
6. Икона Адрес обозначает конец любой ветки, кроме последней.
7. Ветки позволяют
  - разбить медицинский алгоритм на смысловые части,
  - дать частям удобные и точные названия.
8. Шапка — верхняя часть силуэта, которая включает Заголовок алгоритма и комплект икон «Имя ветки».
9. Шапка позволяет читателю быстро сориентироваться в задаче и, расчленив ее на части, увидеть структуру алгоритма.
10. Ветки нужно упорядочить слева направо по принципу: «Более правая ветка работает позже, чем любая ветка, находящаяся левее нее» (кроме веточных циклов).

## Глава 12

# СЛОЖНЫЕ МЕДИЦИНСКИЕ АЛГОРИТМЫ. СИЛУЭТ. ПРАВИЛА И ПРИМЕРЫ

## МЕДИЦИНСКИЕ ПРИМЕРЫ

В прошлой главе мы получили общее представление о силуэте. В этой главе речь пойдет о применении силуэта в медицине.

Алгоритмическая конструкция силуэт позволяет четко описать наиболее сложные особенности медицинской деятельности, включая тончайшие нюансы и мельчайшие подробности. Силуэт способен удовлетворить разнообразные требования и пожелания врачей, возникающие при разработке и использовании сложных и сверхсложных медицинских алгоритмов.

## АЛГОРИТМ СИЛУЭТ «СНЯТИЕ ШЛЕМА С МОТОЦИКЛИСТА»

Для начала рассмотрим вопрос: как превратить примитив в силуэт?

Возьмем живой пример и превратим уже знакомый нам примитив на рис. 82 в силуэт, изображенный на рис. 94.

В чем недостаток примитива? Он не показывает структуру алгоритма. В самом деле, примитив на рис. 82 можно разбить на две части:

- Обеспечение стабильности головы.
- Снятие шлема.

Можно ли заметить эти части, глядя на рис. 82? Нет, нельзя. Нужно долго и мучительно соображать, чтобы догадаться, что граница между частями проходит между вторым и третьим синхронными участками (ниже иконы «Рукой охвати затылок»).

В силуэте на рис. 94 картина радикально меняется. В верхней части появились четкие указатели — две новые иконы «Имя ветки». Словно яркие рекламные заголовки, они предъявляют глазу названия структурных частей алгоритма.

## КАК ЧИТАТЬ АЛГОРИТМ СИЛУЭТ

На рис. 94 надписи нужно читать сверху вниз вдоль левого шампуря (жирная линия). Шампур кончается в иконе Адрес «Снятие шлема».

Куда идти дальше? Подсказка живет внутри самой иконы Адрес. По правилам ДРАКОНа такое же название должно быть наверху — в одном из рекламных заголовков.

Переведите взгляд наверх, найдите икону «Снятие шлема» и спуститесь вниз по правому шампуру до конца.

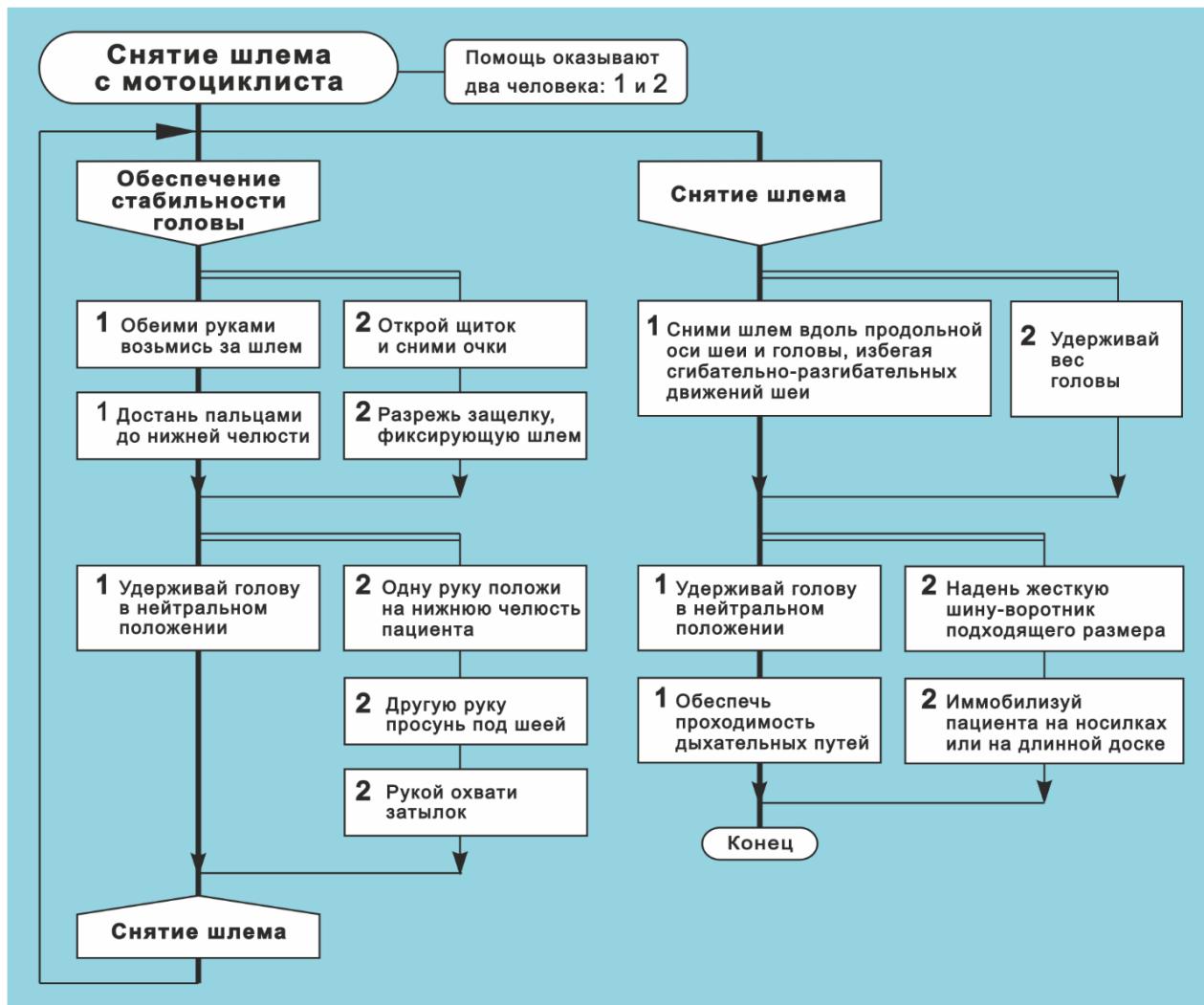


Рис. 94. Алгоритм силуэт, описывающий работу двух врачей [141]

## РАЗДЕЛИЙ И ВЛАСТВУЙ. ВЕТКИ ОБЛЕГЧАЮТ ПОНИМАНИЕ

Наша цель — облегчить зрительное восприятие и анализ сложных медицинских алгоритмов. Спросим себя: можно ли сделать алгоритм обозримым и удобным для быстрого запечатления, обдумывания и осмысления?

Можно. Чтобы любой врач смог легко прочитать и быстро понять алгоритм, надо заблаговременно разделить его на смысловые части. Сделать это нетрудно — надо использовать конструкцию силуэт. Преимущество в том, что силуэт заранее нарезан на удобные ломтики, предназначенные для быстрого схватывания и понимания.

Каждый такой ломтик (ветка) представляет собой довольно крупный кусок алгоритма. В составе ветки обычно несколько икон.

На рис. 94 две ветки, одна слева, другая справа. Первая озаглавлена как «Обеспечение стабильности головы», она содержит 10 икон. Вторая носит имя «Снятие шлема» и имеет 8 икон.

Каждая ветка имеет свой шампур.

## АЛГОРИТМ СИЛУЭТ «ПЕРВАЯ ПОМОЩЬ ПРИ ХИМИЧЕСКОМ ОЖОГЕ ГЛАЗ ЖИДКОСТЬЮ»

Перейдем к следующему примеру (рис. 95, 96). Это алгоритмический рассказ, в котором можно выделить три части, три самостоятельных темы:

1. Промывание глаз водой.
2. Промывание глаз нейтрализующим раствором.
3. Лекарственная обработка.

Каждая тема нарисована в виде ветки.



Рис. 95. Шапка раскрывает структуру силуэта на рис. 96

## ЦАРСКИЕ ВОПРОСЫ

Вспомним наши царские вопросы:

1. Как называется задача?
2. Из скольких частей она состоит?
3. Как называется каждая часть?

Сложный алгоритм можно сравнить с лабиринтом. Чтобы не заблудиться, нужен план лабиринта. Шапка и есть такой план, путеводитель по лабиринту. Изучая шапку, получим ответы для алгоритма на рис. 96.

— Как называется задача? (*Читаем Заголовок алгоритма*).

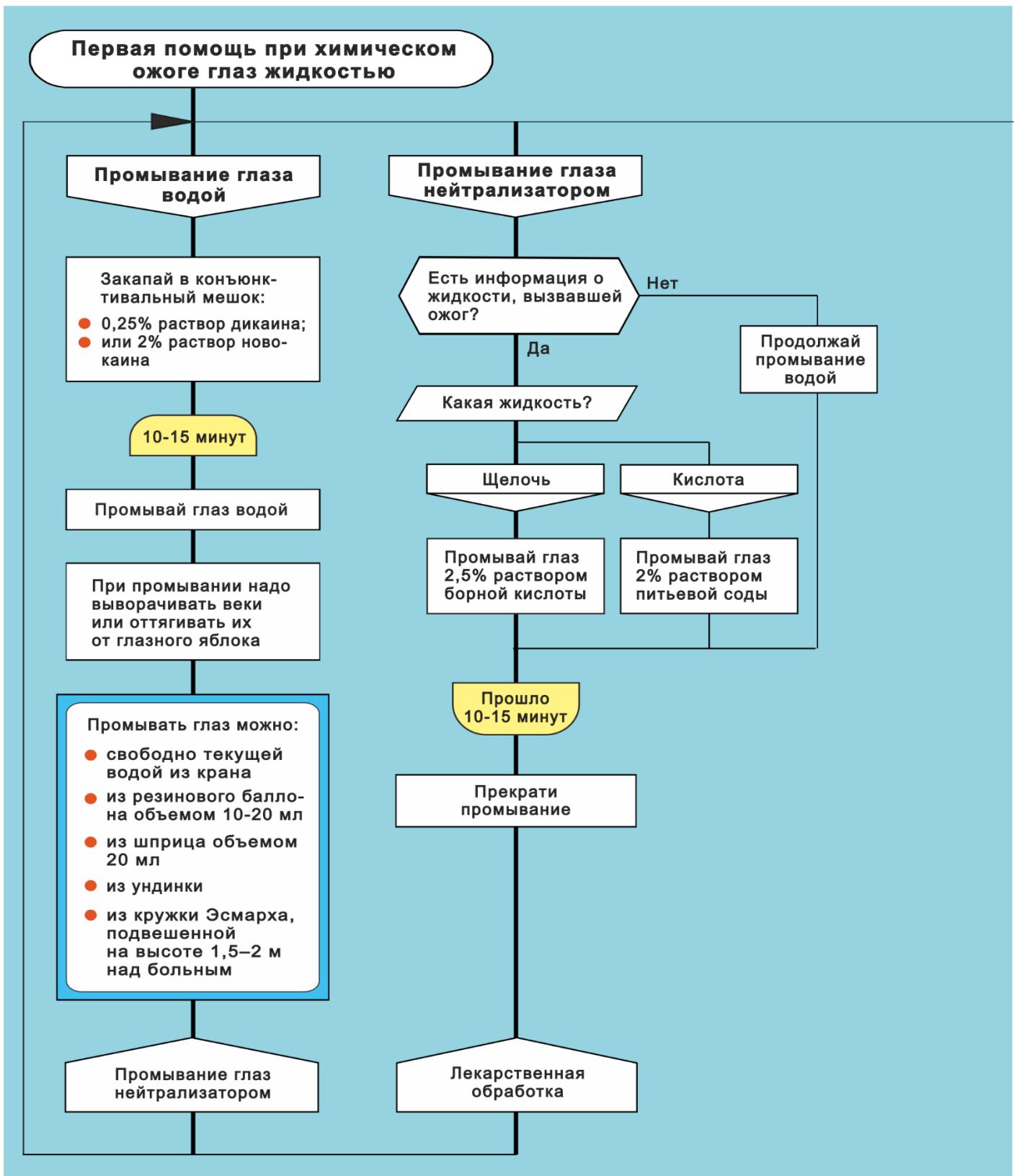
Первая помощь при химическом ожоге глаз жидкостью.

— Из скольких частей она состоит? (*Считаем иконы «Имя ветки»*).

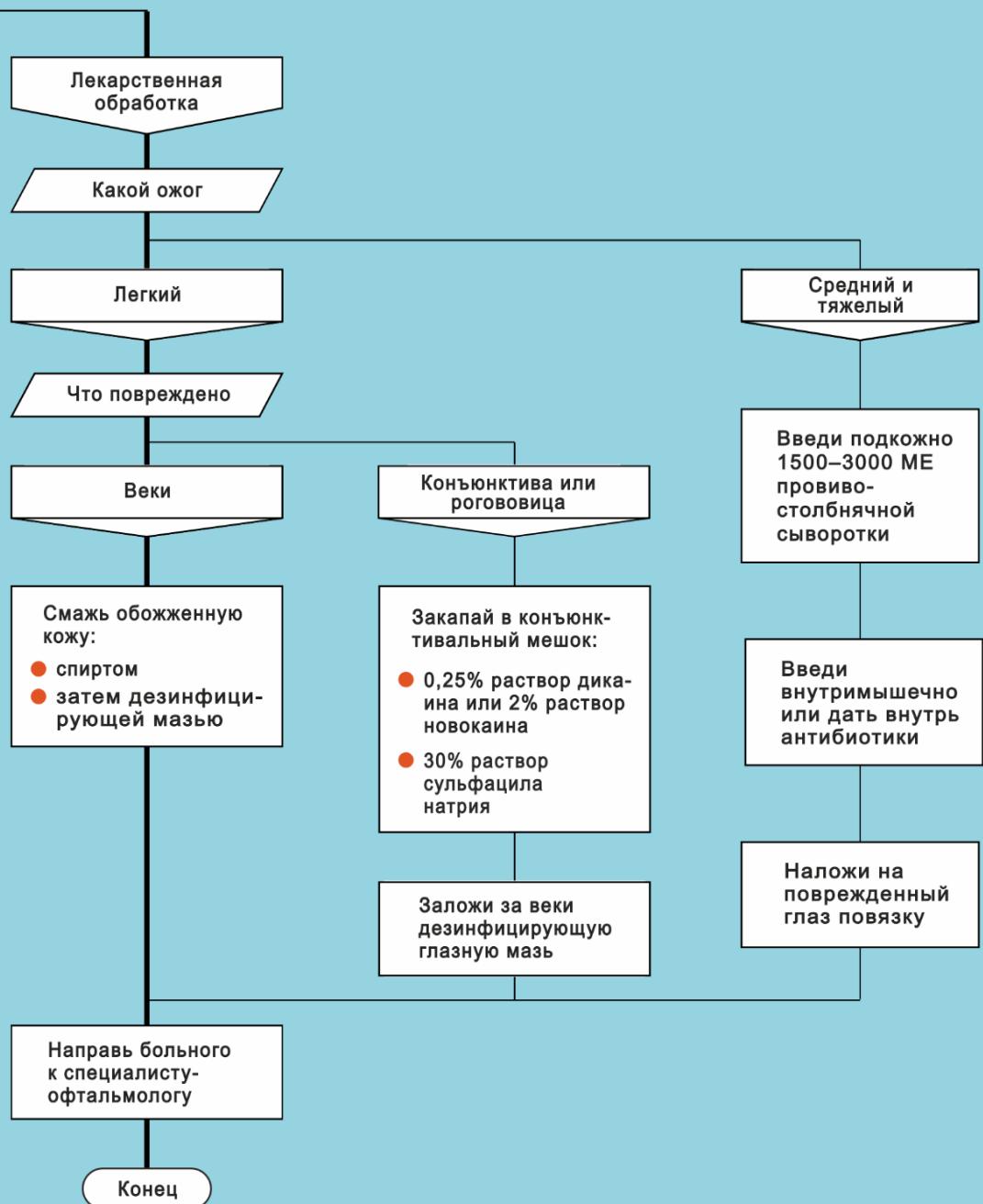
Из трех.

— Как называется каждая часть? (*Читаем текст в иконах «Имя ветки»*).

1. Промывание глаз водой.
2. Промывание глаз нейтрализующим раствором.
3. Лекарственная обработка.



**Рис. 96.** Алгоритм «Первая помощь при химическом ожоге глаз жидкостью»



**Рис. 96.** Алгоритм «Первая помощь при химическом ожоге глаз жидкостью»

## ШАПКА ПРИКОВЫВАЕТ К СЕБЕ ВНИМАНИЕ

Дополнительные удобства связаны с тем, что шапка занимает «парадное» место в верхней части чертежа. Названия смысловых частей помещены внутри особых рамок уникальной формы, которые легко отыскать взглядом.

Благодаря этому шапка моментально приковывает к себе внимание читателя без всяких усилий с его стороны. Поэтому человеку не приходится рыскать глазами по темным закоулкам алгоритма, пытаясь выудить нужную информацию.

## ЧИТАЕМ ПЕРВУЮ ВЕТКУ

*Вопрос.* Где начало и конец у первой ветки на рис. 96?

*Ответ.* Начало — икона «Промывание глаз водой». Конец — «Промывание глаз нейтрализующим раствором». Между началом и концом размещается тело ветки, которое содержит пять икон:

- «Закапай в конъюнктивальный мешок...».
- «10-15 минут».
- «Промывай глаз водой».
- «При промывании надо выворачивать веки...».
- «Промывать глаз можно...».

*Вопрос.* Зачем нужна икона Адрес «Промывание глаз нейтрализующим раствором»?

*Ответ.* Икона Адрес говорит: «Прыгни в икону “Имя ветки” с таким же названием». Смысл в том, что после выполнения ветки «Промывание глаз водой» будет выполняться ветка «Промывание глаз нейтрализующим раствором».

## ЧИТАЕМ ВТОРУЮ ВЕТКУ

Вторая ветка разветвленная, в ней три маршрута. Прочитаем главный из них, который идет по шампуру (рис. 96).

Начало ветки — икона «Промывание глаз нейтрализующим раствором». Конец — «Лекарственная обработка». Между началом и концом находятся иконы:

- «Есть информация о жидкости...?» Да.
- «Какой маршрут?».
- «Щелочь».
- «Промывай глаз 2,5% раствором...».
- «Прошло 10-15 минут».
- «Прекрати промывание».

В самом низу виднеется икона Адрес «Лекарственная обработка». Она говорит: «Прыгни в икону “Имя ветки” с таким же названием». Это значит, что после выполнения ветки «Промывание глаз нейтрализующим раствором» будет выполняться следующая ветка — «Лекарственная обработка».

Оставшиеся два маршрута выполняются аналогично.

## ЧИТАЕМ ТРЕТЬЮ ВЕТКУ

Третья ветка также имеет три маршрута. Пройдемся по главному маршруту (рис. 96).

В начале ветки лежит икона «Лекарственная обработка». В самом низу икона «Конец» — делу венец. Между началом и концом нарисованы шесть элементов:

- «Какой ожог».
- «Легкий».
- «Что повреждено».

- «Веки».
- «Смажь обожженную кожу».
- «Направь больного к специалисту».

Боковые маршруты третьей ветки мы пропускаем, поскольку они очень просты.

## ДРУГОЙ СПОСОБ ОПИСАНИЯ СИЛУЭТА

Силуэт можно описывать двумя способами. Выше мы продемонстрировали метод обычного чтения.

Второй способ — метод бегунка. Вспомним, что воображаемый бегунок при работе алгоритма пробегает алгоритмическую дорожку от иконы Заголовок до иконы Конец.

Как работает алгоритм на рис. 96? Выехав из иконы Заголовок, бегунок мчится вниз по крайней левой ветке. Он движется через станции:

- «Промывание глаз водой».
- «10-15 минут».
- и т. д.

Затем он попадает в икону Адрес — последнюю станцию первой ветки. Куда ехать дальше? Ответ прячется в самой иконе. Эта икона потому и зовется «Адрес», что сообщает адрес (название) следующей станции. Бегунок узнает, что он должен попасть на станцию «Промывание глаз нейтрализующим раствором».

Данная станция находится в начале второй ветки. Но как туда попасть? По какой линии?

Ответ известен. Выехав из иконы Адрес, бегунок резко берет влево и по внешней обрамляющей линии огибает всю схему. Потом едет направо по стрелке и попадает в верхнюю икону «Промывание глаз нейтрализующим раствором».

Дальше события развиваются, как обычно. Бегунок скользит вниз по второй ветке. Добравшись до иконы Адрес, узнает адрес нужной станции («Лекарственная обработка»). Затем огибает схему по часовой стрелке, попадает в начало третьей ветки и спускается по ней до самого конца. На этом наш рассказ завершается.

Правило силуэта

- Любой алгоритм силуэт имеет только одно начало (икону Заголовок). И только один конец (икону Конец).
- Все маршруты силуэта начинаются в иконе Заголовок и кончаются в иконе Конец.

## КОНТРОЛЬНОЕ ВРЕМЯ ПРОЦЕДУРЫ

Многие медицинские действия и решения привязаны ко времени. Типичным примером является так называемое контрольное время. На рис. 96–98 оно показано в виде двух икон желтого цвета:

- Начало контрольного срока.
- Конец контрольного срока.

В руководстве длительность промывания глаз при химическом ожоге строго определена:

«Промывание водой или нейтрализующими растворами продолжают в течение 10–15 минут, чтобы удалить из конъюнктивального мешка ту



Рис. 97. Икона «Начало контрольного срока»

часть повреждающего агента, которая еще не вступила в соединение с тканями» [143].

В алгоритме указанную длительность рассматривают как **контрольное время**, которое должно быть задано в виде двух точно определенных моментов времени: начального и конечного. При этом следует различать два случая:

- моменты находятся в разных ветках,
- оба момента расположены в одной ветке.

В первом случае используют иконы «Начало контрольного срока» и «Конец контрольного срока».

В нашем примере так и сделано. Икона «Начало контрольного срока» поставлена в первой ветке перед иконой «Промывай глаз водой» (рис. 96, 97). А «Конец...» во второй ветке выше иконы «Прекрати промывание» (рис. 96, 98).

Учтите зрительную подсказку: икона Начало похожа на верхнюю полусферу, икона Конец — на нижнюю полусферу.

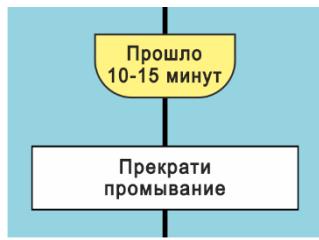


Рис. 98. Икона «Конец контрольного срока»

## МАРШРУТЫ ВЕТКИ

Что такое шампур ветки

Это вертикаль, соединяющая икону «Имя ветки» с иконой Адрес, а если ветка имеет несколько икон Адрес — с левой из них

Каждая ветка силуэта имеет свой шампур. На рис. 96 три ветки. Следовательно, данный силуэт имеет три шампура (три жирных вертикали).

Для ветки сохраняют силу оба царских правила:

- главный маршрут ветки должен идти по шампуру ветки;
- все маршруты ветки следует упорядочить по принципу: «Чем правее, тем хуже».

Последнее правило поясним на примере ветки «Лекарственная обработка» на рис. 96. Она имеет три вертикали.

Левая вертикаль (главный маршрут ветки) описывает наименьший ущерб для пациента, так как ожог легкий, причем обожжены только веки, а наиболее нежные ткани (конъюнктива и роговица) не пострадали.

Средняя вертикаль занимает промежуточное положение. Речь идет о легком ожоге, но, к сожалению, обожженными оказались конъюнктива и роговица.

Правая вертикаль означает наихудшую ситуацию, описывающую средний или даже тяжелый химический ожог.

Правило маршрутов ветки

Чем правее (чем дальше от шампура данной ветки) расположена очередная вертикаль, тем более неприятную для пациента ситуацию она описывает

Вопрос. Сколько маршрутов в силуэте на рис. 96?

Ответ. Во второй ветке 3 маршрута, в третьей тоже 3. Следовательно, общее число маршрутов в силуэте равно  $3 \times 3 = 9$ .

Как выполняются действия внутри ветки?

Действия внутри ветки выполняются по принципу: «Чем ниже, тем позже» (при отсутствии циклов)

## СВОЙСТВА ВЕТКИ

В силуэте на рис. 96 три ветки. Каждая ветка имеет следующие свойства.

Вверху врач видит начало ветки, внизу — конец ветки.

Вход в ветку возможен только через ее начало.

Выходом из ветки служит икона Адрес, в которой записывается имя следующей по порядку исполнения ветки.

Выход из последней ветки осуществляется через икону Конец.

## ЧТО БУДЕТ, ЕСЛИ УБРАТЬ ОБРАМЛЕНИЕ?

Понятие *обрамление* в силуэте включает четыре элемента:

- нижнюю горизонталь,
- верхнюю горизонталь,
- левую вертикальную линию, соединяющую их,
- маленькую стрелку в левом верхнем углу.

Уберем обрамление из алгоритма на рис. 96 и представим полученный чертеж на рис. 99.

Мы видим удивительную картину. Целостный алгоритм распался на три изолированные части, похожие на три острова в океане. Связи между ними полностью исчезли! Острова разделены четкими синими проливами, причем мосты между островами отсутствуют.

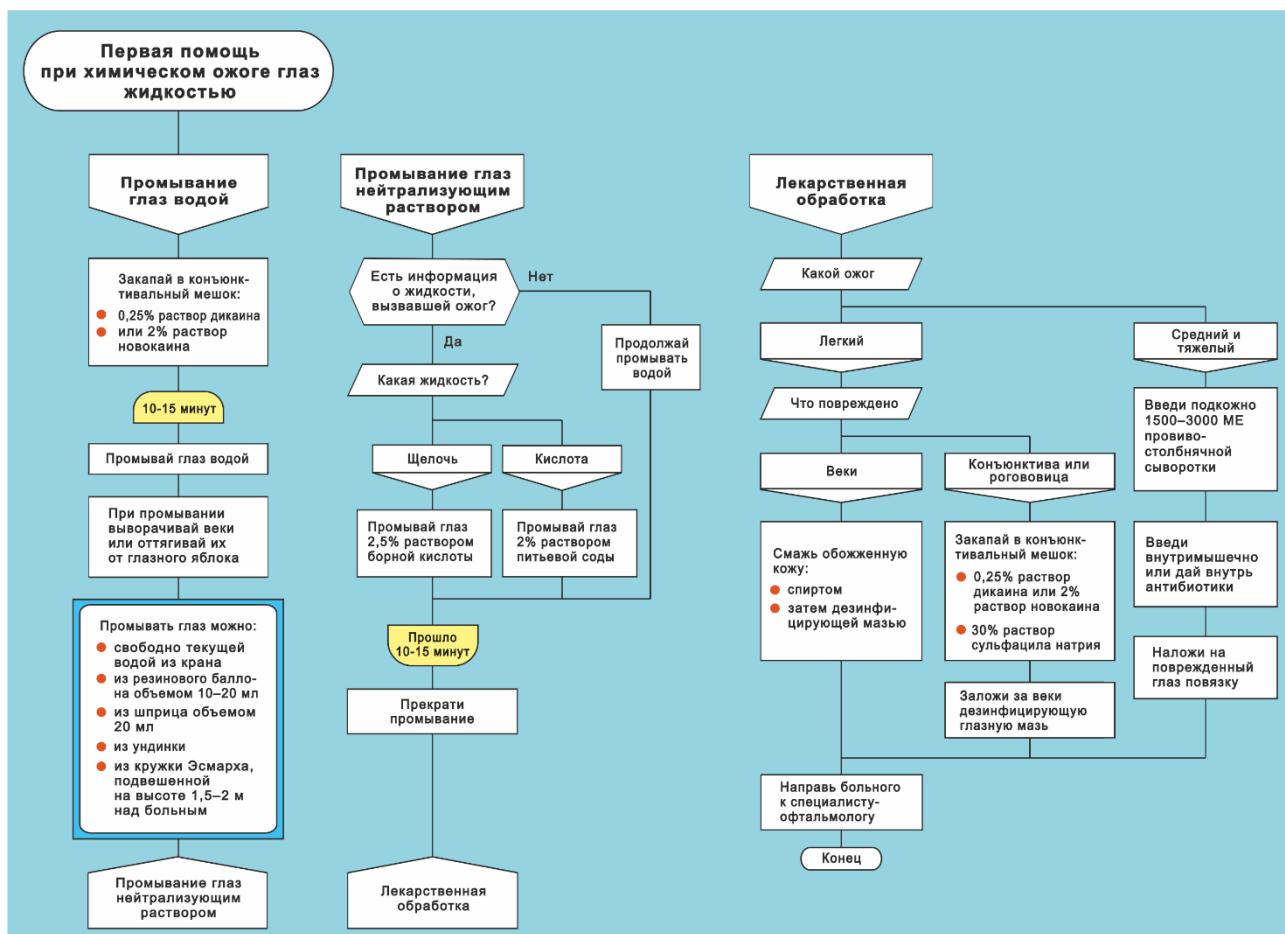


Рис. 99. Алгоритм с удаленными обрамляющими линиями. Сравните с рис. 96

Мы знаем, однако, что все эти украшения носят чисто художественный, декоративный, эстетический характер. Отсутствие обрамления никак не влияет на работу алгоритма. Причина нам также хорошо известна. Взаимодействие между ветками силуэта объясняется не художественными изысками обрамления, а невидимыми логическими связями между «островами». В качестве таких связей выступают указания, записанные в иконах Адрес. Именно эти указания являются невидимым kleem, который скрепляет между собой разрозненные ветки силуэта. И превращает их в единый монолитный, слаженно действующий ансамбль.

Рис. 99 помогает уяснить две истины:

- Связи между ветками возможны только через иконы Адрес.
- Любые иные связи между ветками запрещены.

Смысл запрета в том, что соединительные линии между ветками объявляются незаконными. Если кто-то проведет явную линию между ветками (отличную от линий обрамления), это рассматривается как грубая ошибка. Такая ошибка кажется незаметной на рис. 96. Но на рис. 99 она будет сразу выявлена — как попытка построить пиратский мост между островами.

Завершая этот параграф, следует подчеркнуть, что линии обрамления нужны по эргономическим соображениям. Они, словно рама у картины, соединяют, сочленяют разрозненные части зрительной сцены. Они превращают их в приятный для глаза целостный зрительно-смысловый образ. Исключение скрепляющих линий недопустимо, оно приводит к негативным последствиям — схема зрительно рассыпается на части, что сбивает с толку читателя (рис. 99).

## ВЫВОДЫ

1. Алгоритм силуэт позволяет описать наиболее сложные особенности медицинской деятельности.
2. Примитив не показывает структуру, а силуэт наглядно демонстрирует структурные части алгоритма.
3. Силуэт позволяет облегчить зрительное восприятие и анализ сложных медицинских алгоритмов.
4. Силуэт заранее расченен на удобные структурные элементы (ветки), предназначенные для быстрого схватывания и понимания.
5. Все маршруты силуэта начинаются в иконе Заголовок и кончаются в иконе Конец
6. Если границы контрольного времени находятся в разных ветках, используются иконы «Начало контрольного срока» и «Конец контрольного срока».
7. Шампур ветки — вертикаль, соединяющая икону «Имя ветки» с иконой Адрес, а если ветка имеет несколько Адресов, — с левой из них
8. Главный маршрут ветки должен идти по шампуру ветки.
9. Чем правее (чем дальше от шампера данной ветки) расположена очередная вертикаль, тем более неприятную для пациента ситуацию она описывает.
10. Действия внутри ветки выполняются по принципу: «Чем ниже, тем позже» (при отсутствии циклов).
11. Связи между ветками возможны только через иконы Адрес. Все остальные соединения между ветками запрещены.

## Глава 13

# КАРТОГРАФИЧЕСКИЙ ПРИНЦИП МЕДИЦИНСКОГО АЛГОРИТМА (ПРИНЦИП КРАСОТЫ)

### ЧТО ДУМАЮТ УЧЕНЫЕ О КРАСОТЕ. КРАСОТА КАК ЭРГОНОМИЧНОСТЬ

Понятие «красота» в науке можно трактовать по-разному. Существует мнение, что красота — это совершенство научных результатов, вызывающее восхищение у научного сообщества. Такое восхищение чем-то сродни преклонению перед волшебством и тайнами поэзии. «Нельзя быть математиком, не будучи в то же время и поэтом в душе», — считает немецкий математик Карл Вейерштрасс [144].

Несколько иначе понимает красоту известный физик Поль Дирак:

«Общие законы природы, когда они выражены в математической форме, обладают математической красотой... Это дает физику-теоретику могучий метод, руководящий его действиями. Если он видит, что в его теории есть уродливые части, то он считает, что именно эти части неправильны и он должен сконцентрировать на них свое внимание [чтобы их улучшить]... Этот прием изыскания математического изящества является наиболее существенным для теоретиков» [145].

Иногда говорят, что красота отражает эмоциональную, психологическую восторженность человека, вызванную наличием гармонии, изящества, неординарности в продуктах научного творчества [146].

Но есть и другая точка зрения. Физик Леон Николя Бриллюэн полагает, что красота выражает «приятное ощущение понимания и постижения». Для наших целей позиция Бриллюэна представляет большой интерес, так как она позволяет трактовать красоту как эргономичность.

## КРАСОТА АЛГОРИТМОВ

Алгоритм, представленный в письменном виде, предназначен для зрительного восприятия человеком. Следовательно, алгоритм представляет собой зрительную сцену. Или, если угодно, зрительный образ, зрительную картину.

Красота алгоритма — это красота его зрительного образа, красота дракон-схемы.

Алгоритм можно назвать *красивым* (эргonomичным), если процесс зрительного восприятия, понимания и постижения алгоритма протекает с максимальной скоростью, наименьшими усилиями и максимальным эстетическим наслаждением.

Чем красивее алгоритм, тем быстрее и легче можно его понять. Отсюда вытекает, что красота и элегантность алгоритмов открывают путь к экономии умственных усилий. Но не только.

Чем красивее зрительные образы отдельных частей алгоритма, чем изящнее они соединены в общую алгоритмическую картину, тем приятнее на них смотреть. Чем точнее и элегантнее зрительный «пейзаж» обнажает глубинный смысл алгоритма, тем плодотворнее мышление.

Чем больше красоты, тем глубже понимание алгоритмов. Тем скорее течет наша алгоритмическая мысль. Тем легче мы постигаем суть дела. Тем быстрее и качественнее протекает процесс создания медицинских алгоритмов.

И наоборот, если зрительный образ алгоритма кажется некрасивым, неприятным, отталкивающим и запутанным, процесс понимания и обдумывания неизбежно замедляется, что снижает производительность умственного труда.

ДРАКОН — графический язык, язык зрительных образов. С учетом сказанного, можно уточнить: ДРАКОН — язык красивых (эргономичных) зрительных образов. Но красота ДРАКОНа — не самоцель. Она позволяет ощутимо повысить производительность труда при разработке, изучении и понимании медицинских алгоритмов.

Мы исходим из того, что зрительные образы алгоритмов следует сознательно проектировать. Для этой цели можно (в разумных пределах) использовать *средства художественного конструирования*.

Уместно напомнить слова видного психолога Бориса Ломова (создателя Института психологии АН СССР):

«Средства художественного конструирования в конечном счете направлены на то, чтобы вызвать тот или иной эффект у работающего человека...

Применяя средства художественного конструирования, мы создаем положительные эмоции, облегчаем операцию приема информации человеком, улучшаем концентрацию и переключение внимания, повышаем скорость и точность действий.

Короче говоря, мы пользуемся этими средствами для управления поведением человека в широком смысле слова, для управления его психическим состоянием» и умственной работоспособностью» [199].

Это небольшое вступление, посвященное красоте и эргономичности научных знаковых систем, призвано пролить дополнительный свет на основную идею силуэта,

которая представляет собой принцип карты, или картографический принцип. При дальнейшем изложении мы избегаем рассуждений о красоте, однако ее отблески, несомненно, подразумеваются.

## КАРТОГРАФИЧЕСКИЙ ПРИНЦИП СИЛУЭТА

Картографический принцип говорит о том, что движение взора по географической карте имеет смысл. Глаза учитывают и отслеживают два направления: север – юг и запад – восток.

Точно так же перемещение взгляда по алгоритму должно иметь строго определенный смысл. Движение глаз вверх-вниз (по вертикалам) и влево-вправо (по горизонтали) не должно быть безразличным и бессмысленным. Наоборот, оно должно нести важную информацию и, проникая через хрусталик глаза, обогащать наш мозг цennыми сведениями.

В главе 7 мы описали картографический принцип примитива. Он означает, что слева находятся более благоприятные для пациента (хорошие) маршруты, справа — менее благоприятные (плохие). По вертикалам смысл такой: вверху начало времени, внизу — конец.

Следует учесть, что примитив — аналог ветки. Следовательно, картографический принцип примитива полностью сохраняет силу для ветки.

Оговоримся. Принцип для ветки имеет локальный характер; он действует внутри одной ветки. И не распространяется за ее пределы.

Сделаем следующий шаг и определим порядок взаимодействия веток. И тем самым распространим картографический принцип на весь силуэт.

Для этого нужно упорядочить ветки в пространстве. Лучше всего расположить их слева направо в той последовательности, в какой они включаются в работу. Для этого служит правило: «Чем правее, тем позже». Оно означает: чем правее находится ветка, тем позже она работает.

Правило  
построения веток  
в силуэте

Чтобы силуэт был красивым и удобным для работы, ветки нужно расположить слева направо согласно принципу: «Более правая ветка работает позже, чем любая ветка, находящаяся левее нее» (кроме веточных циклов).

Силуэт, нарисованный согласно правилу «Чем правее, тем позже», считается красивым, эргономичным (рис. 96). Схемы, где это правило нарушается, объявляются плохими. Их использование запрещено.

В разрешенных (эргономичных) алгоритмах имеет место следующий порядок действий (рис. 96):

- первой работает крайняя левая ветка, последней — крайняя правая;
- остальные ветки передают эстафету друг другу слева направо (при этом может случиться так, что некоторые ветки будут пропущены).

Отсюда следует, что время в силуэте перемещается не по одной, а по двум осям — и по вертикалам, и по горизонтали.

- Двигаясь по ветке сверху вниз, мы перемещаемся во времени от начального момента до конечного (на данном отрезке). Время движется по ветке вертикально вниз.
- Двигаясь по веткам слева направо, мы тоже перемещаемся во времени. По разным веткам время движется горизонтально слева направо (при отсутствии веточных циклов).

При этом никакой путаницы не возникает, все учтено и хорошо продумано в соответствии с принципом красоты. По одной ветке время бежит всегда вниз, по разным веткам — всегда направо (кроме циклов).

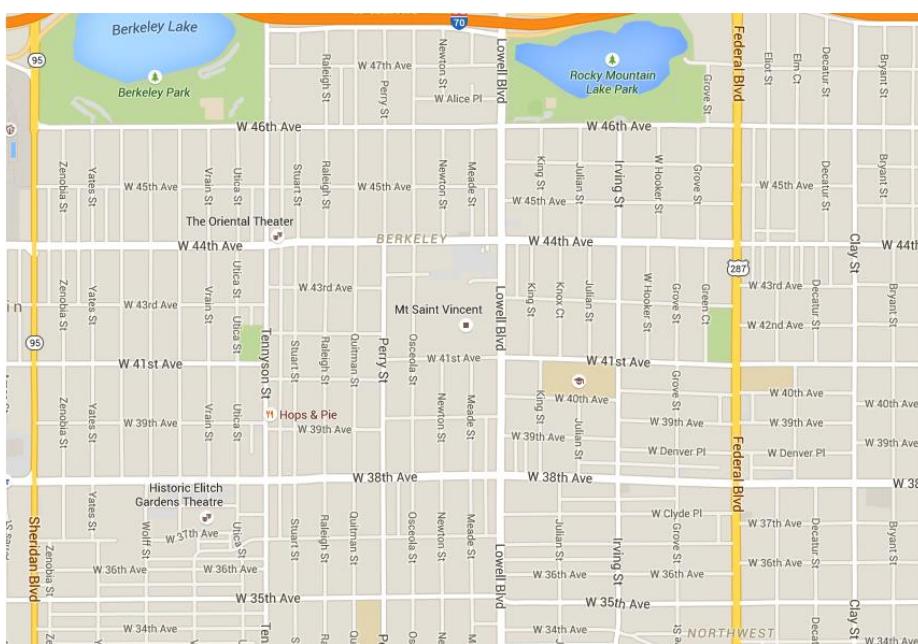
## МОЖНО ЛИ НАВЕСТИ ПОРЯДОК В МЕДИЦИНСКИХ АЛГОРИТМАХ?

В некотором царстве, в тридесятом государстве были два города. Один строился без плана, вкривь и вкось: тесные улочки и переулки сплелись в змеиный клубок (рис. 100). Зато другой был образцовым: красивые площади, широкие проспекты, всюду простор и порядок (рис. 101). В каком городе легче найти дорогу?

Конечно, во втором.



**Рис. 100.** Хаотическая планировка и запутанная застройка с кривыми улочками характерна для многих старинных городов, которые стихийно разрастались на месте древних поселков



**Рис. 101.** Современное строительство города на равнине позволяет заранее оставить четкий план и строго его соблюдать. Все чинно-благородно, никаких нарушений, никакой партизанщины

А теперь представьте, что нам нужно изучить незнакомый медицинский алгоритм. Если в нем, как в первом городе, нет порядка, он превращается в запутанный лабиринт, к которому ничего нельзя понять. Если же он, как второй город, построен по хорошему плану, ситуация в корне меняется. Про такой город говорят: Красота — все ясно как на ладони!

Чтобы убедиться в этом, давайте посмотрим еще разок на рис. 96 и проведем взглядом по всем вертикалям слева направо. Мы обнаружим не хаос, а строгий порядок. Потому что маршруты нарисованы не случайно, не как левая нога захочет, а по строгим правилам, правилам красоты.

## КРАСИВОЕ И УРОДЛИВОЕ

Вспомним слова Поля Дирака: если в «теории есть уродливые части, то... именно эти части неправильны». Данную мысль можно применить и к медицинским алгоритмам. В восьмой главе мы видели, что операция рокировки легко превращает алгоритмического «урода» в «красавца». Язык ДРАКОН способен существенно улучшить эстетические (эргономические) характеристики медицинских алгоритмов.

## ЧТО ЛУЧШЕ: БЛОК-СХЕМА ИЛИ ДРАКОН-СХЕМА?

Существует и другая точка зрения. Специалисты из Самарского медицинского университета рекомендуют рисовать медицинские алгоритмы в виде блок-схем согласно Государственному стандарту ГОСТ 19.701-90. В защиту своей позиции они приводят следующие аргументы:

«Активное введение в учебный процесс... стандартных блок-схем и описываемых ими алгоритмов действий оправдано по трем причинам.

- Во-первых, студенты, как правило, знакомятся с ними еще в школе, на занятиях по информатике.
- Во-вторых, их активно вводят в практику врачи, защищающиеся по относительно новой для медицины специальности 05.13.01 «Системный анализ, управление и обработка информации».
- В-третьих, они в методическом плане позволяют визуализировать ход принятия решений, его этапы и составляющие, причем делается это стандартными и уже знакомыми студентам символами, что облегчает восприятие как блок-схем, так и учебного материала» [147].

Предложение самарских коллег вызывает обоснованные возражения, так как указанный стандарт является калькой международного стандарта ISO 5807-85 тридцатилетней давности, который отстал от жизни, построен без учета современных когнитивно-эргономических идей и концептуально устарел.

Есть ли замена блок-схемам? Да, есть. В 1996 году Государственный комитет Российской Федерации по высшему образованию включил изучение языка ДРАКОН в программу курса «Информатика» для направлений:

- 510000 — Естественные науки и математика
- 540000 — Образование
- 550000 — Технические науки
- 560000 — Сельскохозяйственные науки [148]

В официальном документе Госкомвзуа «Примерная программа дисциплины „Информатика“» имеются разделы, посвященные языку ДРАКОН<sup>11</sup> [149]. В программе, в частности, говорится:

«Синтез идей информатики и эргономики полезен тем, что процесс алгоритмизации (который во многих случаях требует значительных трудозатрат) становится менее трудоемким и более ясным. Для этого вводится понятие “эргономичный алгоритм”. Излагаются равносильные преобразования алгоритмов, способные улучшить их эргономические характеристики» [150].

## **ДРАКОН-СХЕМА — ЭТО КРАСИВАЯ, ПРАВИЛЬНО ПОСТРОЕННАЯ БЛОК-СХЕМА**

Дракон-схемы построены на основе блок-схем с целью их совершенствования и развития.

Недостаток блок-схем заключается в том, что они не приучают к аккуратности при разработке алгоритма. Ромб можно поставить в любом месте блок-схемы, а от него повести выходы на какие угодно участки. «Так можно быстро превратить алгоритм «в запутанный лабиринт, разобраться в котором через некоторое время не сможет даже сам его автор» [151].

Блок-схемы не позволяют изображать сложные алгоритмы с необходимой полнотой и наглядностью. Чтобы устранить дефект, нужно упорядочить блок-схемы. Упорядоченные блок-схемы называются «дракон-схемы» и подчиняются строгим формальным правилам и правилам эргономичных алгоритмов [70].

В отличие от блок-схем, дракон-схемы пригодны для формализованной записи и автоматического получения исполняемого кода. Запрещено пересечение линий, которое путает читателей и затрудняет понимание алгоритма, устраниены другие некрасивые места. Дракон-схемы позволяют ликвидировать или существенно ослабить недостатки блок-схем.

Дракон-схемы специально сконструированы таким образом, чтобы превратить сложный алгоритм в удобную схему, обеспечивающую быстрое и легкое понимание [152] [153]. По мнению специалистов, благодаря использованию дракон-схем алгоритмы становятся более понятными, доходчивыми, ясными, прозрачными. Эргономичные методы, применяемые в дракон-схемах, существенно улучшают восприятие алгоритмов. Язык ДРАКОН обеспечивает разработку сложных алгоритмов с сохранением наглядности даже для многостраничных схем [154] [155].

## **РЕКОМЕНДАЦИИ АВТОРАМ МЕДИЦИНСКИХ УЧЕБНИКОВ**

В некоторых медицинских учебниках в качестве иллюстраций используются блок-схемы алгоритмов. Это ошибка, которая создает неоправданные трудности для студентов медицинских университетов. Ошибка, вызванная отсутствием достоверной информации. Ошибку, которую необходимо исправить.

---

<sup>11</sup> «Примерная программа дисциплины „Информатика“» одобрена Президиумом совета по информатике Госкомвзуа. Председатель Президиума академик РАН Юрий Журавлев — руководитель Секции прикладной математики и информатики Отделения математических наук РАН, а также заместитель Академика-секретаря Отделения математических наук РАН.

Хотя стандарты на блок-схемы считаются действующими, фактически они давно устарели. С появлением дракон-схем блок-схемы полностью потеряли свое значение, так как они во всех отношениях уступают дракон-схемам [156].

Язык ДРАКОН был разработан, в частности, потому, что традиционные блок-схемы алгоритмов, с эргономической точки зрения, не выдерживают критики. Они напоминают непроходимые джунгли, в которых легко запутаться и сбиться с дороги.

ДРАКОН выгодно отличается тем, что его графический узор подчиняется жестким и тщательно продуманным правилам, которые дисциплинируют мышление, облегчают умственный труд [157].

Важным дефектом блок-схем является недостаток выразительных средств. Алгоритмическая структура силуэт, являющаяся основным и наиболее мощным инструментом языка ДРАКОН, принципиально не может быть выражена на языке блок-схем.

Изложенные в литературе теоретические обоснования, сравнительный анализ дракон-схем и блок-схем, а также многочисленные примеры убедительно доказывают, что блок-схемы не имеют будущего [70] [71] [72].

В связи с этим можно рекомендовать авторам медицинских учебников отказаться от блок-схем и использовать язык ДРАКОН как наиболее эффективное средство для записи медицинских алгоритмов, помогающее облегчить учебный процесс и повысить качество обучения.

## ВЫВОДЫ

1. Медицинский алгоритм называется красивым (эргономичным), если процесс зрительного восприятия и осмысливания алгоритма протекает с максимальной скоростью, наименьшими усилиями и доставляет удовольствие читателю.
2. Красивый алгоритм должен быть упорядоченным, пригодным для быстрого восприятия и запоминания.
3. Примитив — аналог ветки.
4. Картографический принцип примитива действителен для ветки.
5. Картографический принцип ветки гласит:
  - в левой части ветки находятся более благоприятные для пациента вертикальные маршруты, в правой — менее благоприятные.
  - ориентация ветки по вертикали такова: вверху начало времени, внизу — конец.
6. Картографический принцип ветки имеет локальный характер и не распространяется за ее пределы.
7. Картографический принцип силуэта гласит:
  - в каждой ветке время направлено вертикально вниз;
  - внутри ветки вертикальные маршруты упорядочены слева направо от наиболее благоприятных для пациента до наименее благоприятных;
  - в силуэте время движется по горизонтали слева направо;
  - первой работает крайняя левая ветка, последней — крайняя правая.
8. В медицинских алгоритмах на языке ДРАКОН все маршруты тщательно упорядочены и организованы в красивую зрительную сцену, предназначенную для облегчения изучения и запоминания алгоритмов.
9. Блок-схемы алгоритмов безнадежно устарели. Пользоваться ими недопустимо. Вместо них следует использовать дракон-схемы

## Глава 14

# АЛГОРИТМ «РЕАНИМАЦИЯ БЕРЕМЕННОЙ ЖЕНЩИНЫ»

### НОВЫЕ ВОЗМОЖНОСТИ

Мы успешно преодолели подготовительные ступени при изучении ДРАКОНа и забрались почти на самую вершину. Осталось сделать последний рывок и овладеть увлекательной темой, которая называется «Многоадресный силуэт». Имеется в виду силуэт, содержащий многоадресные ветки.

До сих пор мы знакомились с простыми случаями, когда все ветки силуэта имеют только одну икону Адрес (рис. 87, 94, 96). Силуэт с многоадресными ветками обладает новыми и многообещающими возможностями:

- можно изменять последовательность выполнения веток,
- можно выборочно запускать ветки, пропуская одну из них и даже несколько,
- можно повторять выполнение веток, т. е. создавать веточные циклы.

Более широкие возможности позволяют существенно улучшить качество медицинских алгоритмов, сделать их более точными и наглядными.

Далее мы рассмотрим поучительный пример, разъясняющий идею многоадресного силуэта. План изложения такой.

- Сначала взглянем на общую картину силуэта.
- Потом выявим структуру с помощью шапки.
- После этого рассмотрим ветки поочередно слева направо, задерживаясь на всех интересных местах.

## **МНОГОАДРЕСНЫЙ СИЛУЭТ «РЕАНИМАЦИЯ БЕРЕМЕННОЙ ЖЕНЩИНЫ»**

Существует ряд причин, которые могут вызвать внезапную (клиническую, т. е. обратимую) смерть беременной женщины. На рис. 102 показан алгоритм реанимации для этого случая. Силуэт делится на шесть смысловых частей – веток. Каждую ветку читают сверху вниз. Разные ветки изучают слева направо, учитывая указания в иконах Адрес.

Начинать чтение нужно с первой, крайней левой ветки, постепенно перемещаясь направо до конца.

Силуэт называется многоадресным, если он содержит многоадресные ветки. На рис. 102 видно, что вторая и третья ветки являются многоадресными, они имеют по три иконы Адрес. По этой причине силуэт на рис. 102 является многоадресным.

### **ВАЖНЫЕ ВЕЩИ НУЖНО ВЫДЕЛЯТЬ. КАК ЭТО СДЕЛАТЬ?**

На рис. 102 взору зрителя одновременно представлено большое количество информации, буквально глаза разбегаются. Если человек впервые видит столь сложную зрительную сцену, он может испытывать дискомфорт и ощущать неудовольствие. Действительно, зрительная картина содержит 42 иконы и множество соединительных линий. Вся эта лавина информации разом обрушивается на неподготовленного зрителя, и человек поневоле теряется.

Чтобы этого не случилось, в языке ДРАКОН предусмотрены специальные средства, позволяющие управлять движением глаз наблюдателя и концентрировать внимание на ключевых позициях и существенных местах.

К специальным средствам относятся:

- картографический принцип силуэта,
- средства управления восприятием,
- фиксация начала и конца,
- шапка.

### **СРЕДСТВА УПРАВЛЕНИЯ ВОСПРИЯТИЕМ**

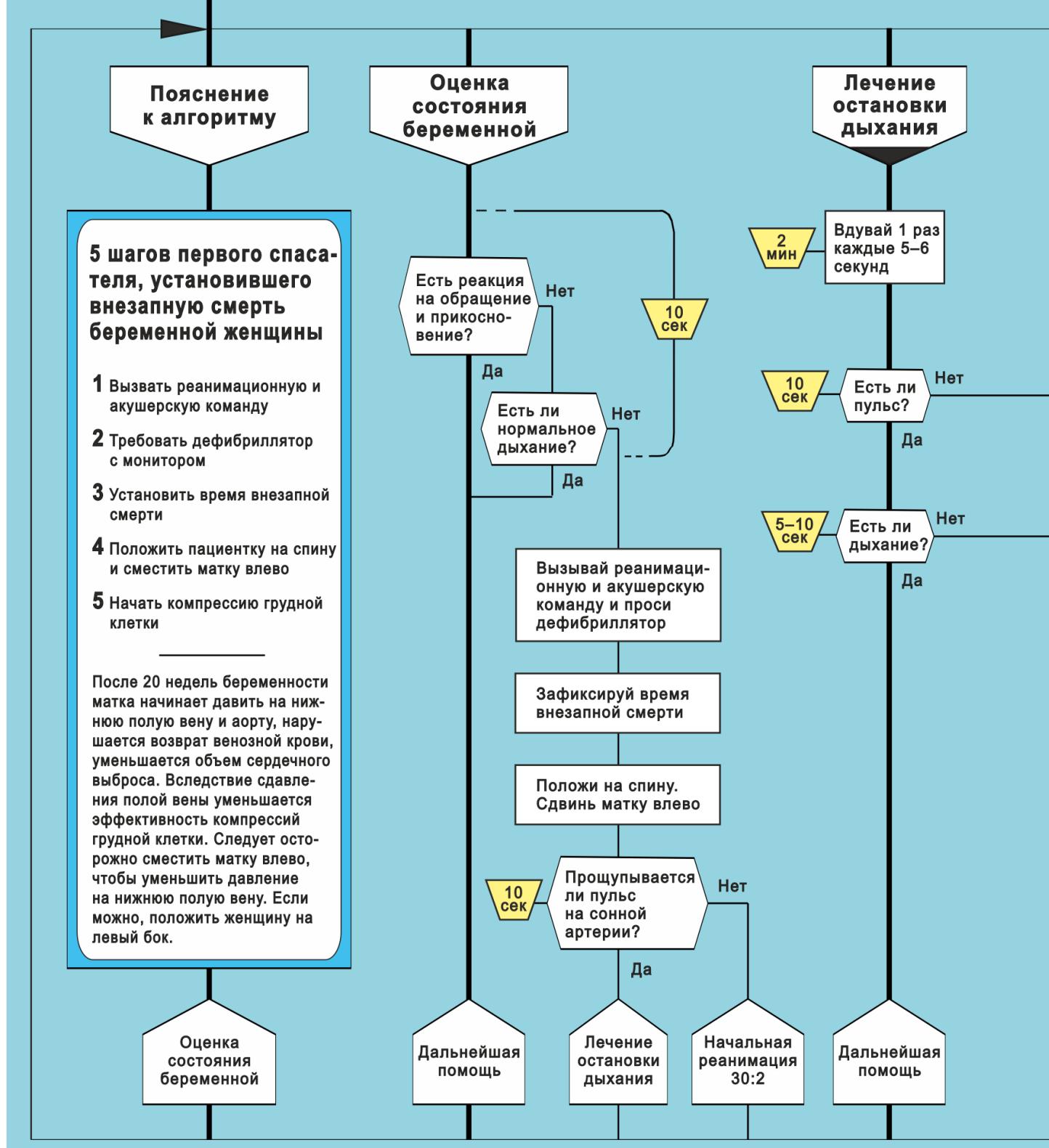
Как привлечь внимание читателя к тем или иным точкам на чертеже алгоритма? Для этого служат средства привлечения внимания, или средства управления восприятием.

Философ Марио Бунге полагает, что интуицию можно, в частности, трактовать как «быстрое восприятие, воображение...» [158]. Интуитивное восприятие осуществляется без слов, без вербализации. Исходя из этого, в языке ДРАКОН используются следующие средства управления восприятием:

- увеличение толщины важных линий,
- выделение шрифтом и кеглем,
- акцент на иконах Время,
- треугольные маркеры,
- ветка-комментарий.

Прелест в том, что человеку не нужно обдумывать действия при переключении внимания на нужные (выделенные) объекты; они как бы сами бросаются в глаза, сами притягивают взгляд. Это происходит потому, что — благодаря умелому

## **Реанимация беременной женщины**



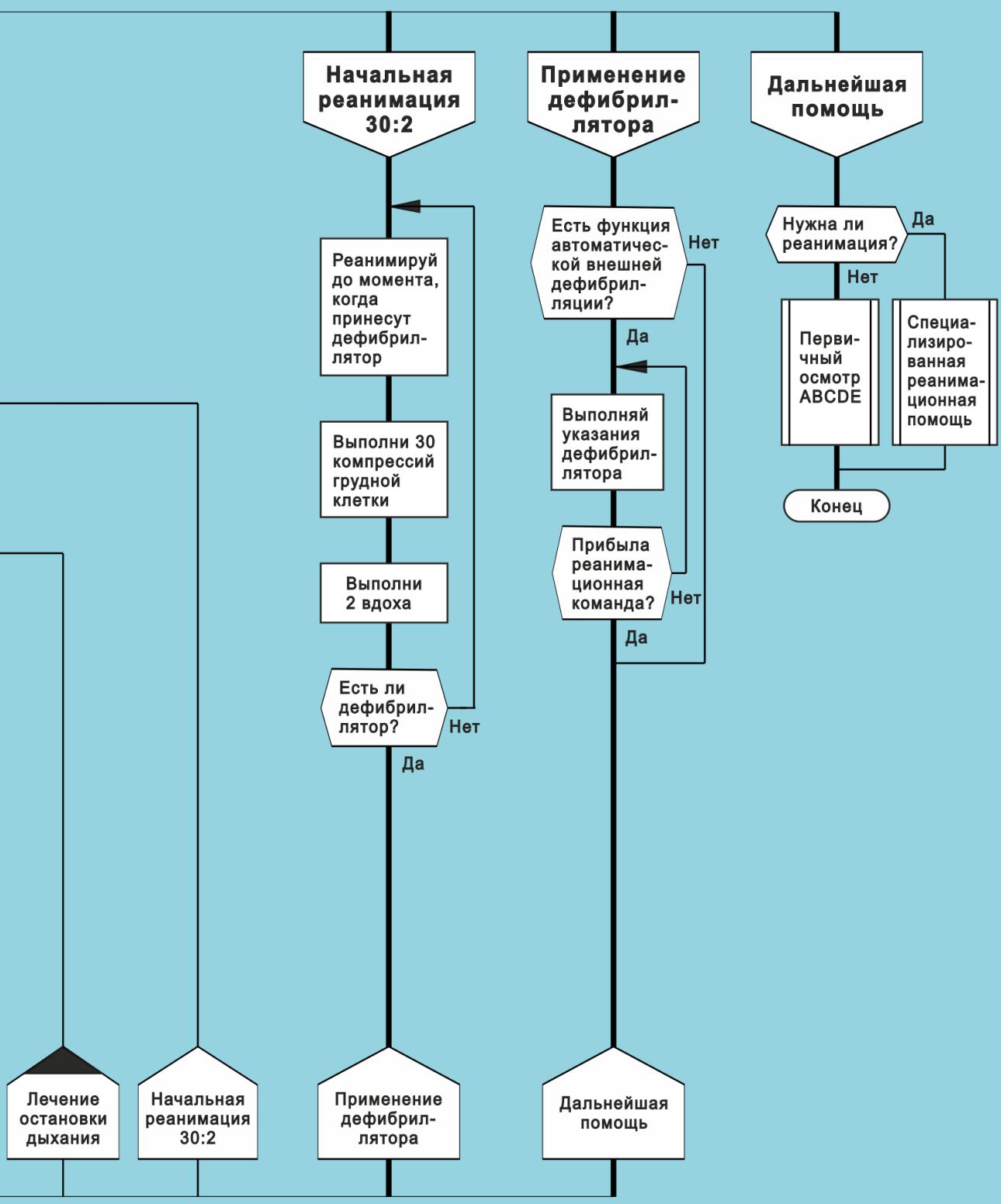


Рис. 102. Алгоритм «Реанимация беременной женщины» [159]

использованию средств управления восприятием — выбранные объекты имеют неоспоримое преимущество перед соседями. Они кажутся более привлекательными и непроизвольно, без участия сознания, попадают в фокус внимания. Перечислим приемы для привлечения внимания на рис. 102.

Основой зрительной композиции являются шампуры. Необходимо выделить шесть шампуром — по одному в каждой ветке. Они сделаны жирными.

Текст в иконах Заголовок и Имя ветки играет важную роль и имеет высший приоритет. Поэтому кегль увеличен и выбран жирный шрифт.

На действия и решения врача накладываются ограничения по времени. Чтобы акцентировать внимание на иконах Время, они выделены желтым цветом.

Два маркера «черный треугольник» позволяют легко заметить веточные циклы.

## ГДЕ НАЧАЛО, ГДЕ КОНЕЦ

Иконы Заголовок и Конец являются пограничными, они обозначают границы алгоритма во времени и пространстве. Все маршруты выходят из Заголовка и собираются в Конце.

Большую роль играет договоренность о месте размещения этих икон. Они должны всегда находиться на одном и том же, заранее известном, фиксированном месте. В блок-схемах такая договоренность отсутствует, что приводит к хаотическому размещению начала и конца, которые прыгают с места на место.

Язык ДРАКОН имеет на этот счет четкие и неизменные правила. В силуэте Заголовок всегда находится в левом верхнем углу. И только там.

Икона Конец также имеет постоянное место — в правом углу, в конце последней ветки.

Почему необходимо точно фиксировать позицию этих икон? Потому, что это облегчает изучение новых и осмысление забытых алгоритмов. Танцевать надо от печки. Проникновение в алгоритм облегчается, если читатель заранее знает, где следует искать две ключевые фигуры: Заголовок и Конец.



Рис. 103. Шапка сложного алгоритма на рис. 102

## ШАПКА ДЛЯ АЛГОРИТМА РЕАНИМАЦИИ

Шапка — двухэтажная фигура, которая задает смысловой костяк алгоритма (рис. 103). Изюминка в том, что шапка делает человеку своевременные подсказки, настраивающие его на нужный лад и облегчающие работу ума.

Прочитаем подсказки для рис. 102, 103.

1. Как называется задача? (Читаем заголовок алгоритма).

- «Реанимация беременной женщины».
2. Из скольких частей она состоит? (*Считаем иконы «Имя ветки»*).
    - Из шести.
  3. Как называется каждая часть? (*Читаем текст в иконах «Имя ветки»*).
    - «Пояснение к алгоритму».
    - «Оценка состояния беременной».
    - «Лечение остановки дыхания».
    - «Начальная реанимация 30:2».
    - «Применение дефибриллятора».
    - «Дальнейшая помощь».

Сколько времени нужно, чтобы впитать эту информацию? Минуту? Две? Три? Пять?

Шапка дает общее представление, без деталей. Благодаря шапке читатель получает ориентировку о проблеме практически сразу. Причем, что очень важно, время ориентировки почти не зависит от сложности задачи и составляет считанные минуты.

Из шапки мы узнали названия веток. После этого, как всегда, приступаем к двум стандартным операциям:

- изучению содержания веток,
- анализу их взаимодействия.

Чтобы выполнить эти операции и тщательно изучить алгоритм на рис. 102, подготовленному читателю вряд ли потребуется больше, чем двадцать тридцать минут. (Речь, разумеется, идет о специалистах или студентах, а не о человеке с улицы).

## ИЗУЧАЕМ ВЕТКУ «ПОЯСНЕНИЕ К АЛГОРИТМУ»

На рис. 102 первая ветка «пустая»; она не выполняет никаких действий. Зачем она нужна?

Рассмотрим случай, когда алгоритм используется в медицинском учебнике в качестве иллюстрации. Обычно его сопровождает развернутый пояснительный текст, расположенный на соседних страницах.

Предположим, авторы учебника заботятся об эргономике и желают сделать материал более дружелюбным. В этом случае можно создать дополнительное удобство для читателей. Надо выделить из пояснительного текста наиболее важные мысли и вставить их непосредственно в чертеж алгоритма в качестве краткого введения.

Для этой цели необходимо:

- добавить в алгоритм еще одну ветку, которая называется «ветка-комментарий» (рис. 104);

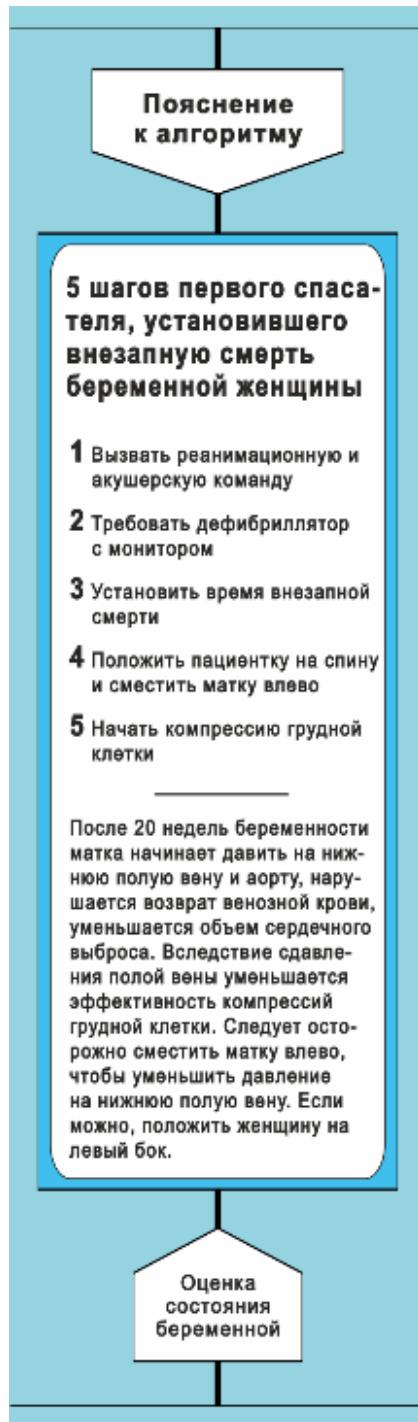


Рис. 104. Ветка-комментарий

- поместить в нее большую икону Комментарий, которая занимает всю ветку;
- записать в Комментарии самые ценные мысли.

По этому рецепту создана крайняя левая ветка на рис. 102. В качестве важных мыслей использованы два избранных отрывка:

- 5 шагов первого спасателя, установившего внезапную смерть беременной женщины [160];
- Информация о процессах, протекающих в организме после 20-й недели беременности, которые могут привести к нарушению возврата венозной крови и уменьшению сердечного выброса [161].

Что мы при этом выиграли? Мы поместили наиболее важную информацию рядом с алгоритмом — на одной зрительной сцене. Тем самым мы избавили читателя от необходимости листать страницы туда-сюда, попеременно воспринимая текст и чертеж, находящиеся на разных страницах. Мы создали дополнительные удобства для читателя, обеспечив снижение нагрузки на его оперативную память.

## ИЗУЧАЕМ ВЕТКУ «ОЦЕНКА СОСТОЯНИЯ БЕРЕМЕННОЙ»

В конце первой ветки в иконе Адрес читаем «Оценка состояния беременной» — это имя следующей ветки, куда следует перейти.

Данная команда переводит наш взгляд (и бегунок) в начало второй ветки. Для удобства чтения ветка выделена из рис. 102 и представлена на рис. 105.

Ветка «Оценка состояния беременной» существенно отличается от тех, что нам уже знакомы. Новинка в том, что она содержит три иконы Адрес. Три разных Адреса указывают на переход в три разные ветки. Зачем это нужно?

Мы знаем, что надписи в иконах Адрес играют роль суфлера и советчика — они однозначно определяют порядок выполнения веток.

В предыдущих примерах (рис. 87, 94, 96) каждая ветка, кроме последней, имела только одну икону Адрес. Поэтому ветки работали последовательно, соблюдая железный порядок: сначала первая, после первой всегда вторая, после второй всегда третья и т. д.

Однако подобная жесткая дисциплина нужна далеко не всегда. Во многих случаях необходимо, чтобы ветка служила «большой



Рис. 105. Ветка «Оценка состояния беременной»

развилкой» и позволяла выбирать различные пути, т. е. указывала на несколько разных веток.

Ветка «Оценка состояния беременной» как раз и выполняет такую функцию. Имея три иконы Адрес, она дает возможность совершить прыжок в три разные ветки:

- «Дальнейшая помощь».
- «Лечение остановки дыхания».
- «Начальная реанимация 30:2».

## ИКОНА «ВРЕМЯ»

Обратите внимание на две маленькие иконы — перевернутые трапеции желтого цвета. Они обозначают Время и выполняют важную функцию — указывают врачу предельное время, отводимое на ту или иную операцию.

Например, верхняя икона «10 секунд» на рис. 105 напоминает, что врач обязан за это время успеть выполнить две операции:

- выяснить, есть ли реакция пациента на обращение и прикосновение,
- выяснить, есть ли у пациента нормальное (не агональное) дыхание.

## ЛИШНЮЮ ИКОНУ СЛЕДУЕТ УДАЛИТЬ

Рассмотрим ситуацию подробнее. Для этого проанализируем икону Время «10 секунд», соединенную с иконой Вопрос «Прощупывается ли пульс на сонной артерии?» (рис. 105).

Икона Вопрос в данном случае несет двойную нагрузку. Во-первых, она обозначает Действие: «Прощупай пульс на сонной артерии». Во-вторых, по результатам Действия врач должен принять Решение: прощупывается пульс или нет?

Поскольку речь идет о реанимации, время дорого и счет идет на секунды. Желтая икона Время напоминает врачу: на обе операции (выполнение Действия и принятие Решения) ему отводится всего 10 секунд.

**Вопрос.** Некоторые люди считают, что любое действие (например, «Прощупай пульс на сонной артерии») следует писать в иконе Действие. Почему в данном случае икона Действие отсутствует на рис. 105?

**Ответ.** Потому что в ней нет никакой необходимости. Если послушать горе-советчиков и добавить ненужную икону (как показано на рис. 106), то надписи в иконах Действие и Вопрос будут почти полностью совпадать:

- «Прощупай пульс на сонной артерии».
- «Прощупывается ли пульс на сонной артерии?»

При восприятии одинакового текста читатель не получает новой информации и зря теряет время. Это может вызвать ненужную заминку и даже раздражение. По этой причине икону Действие удаляют — см. зачеркивание крестом на рис. 106.



Рис. 106. Икона Действие здесь не нужна

Правило лишней иконы

Если две иконы (Действие и Вопрос) соединены последовательно, причем текст в обеих иконах почти полностью повторяется, икона Действие в медицинском алгоритме является лишней. Ее следует удалить

## КАК ИСПРАВИТЬ ОШИБКУ

Алгоритм «Реанимация беременной женщины» мы заимствовали из источника [159]. При этом выявились интересная подробность. В оригинале ветка «Оценка состояния беременной» изображена, как показано на рис. 107. Обратите внимание на порядок икон Адресс:

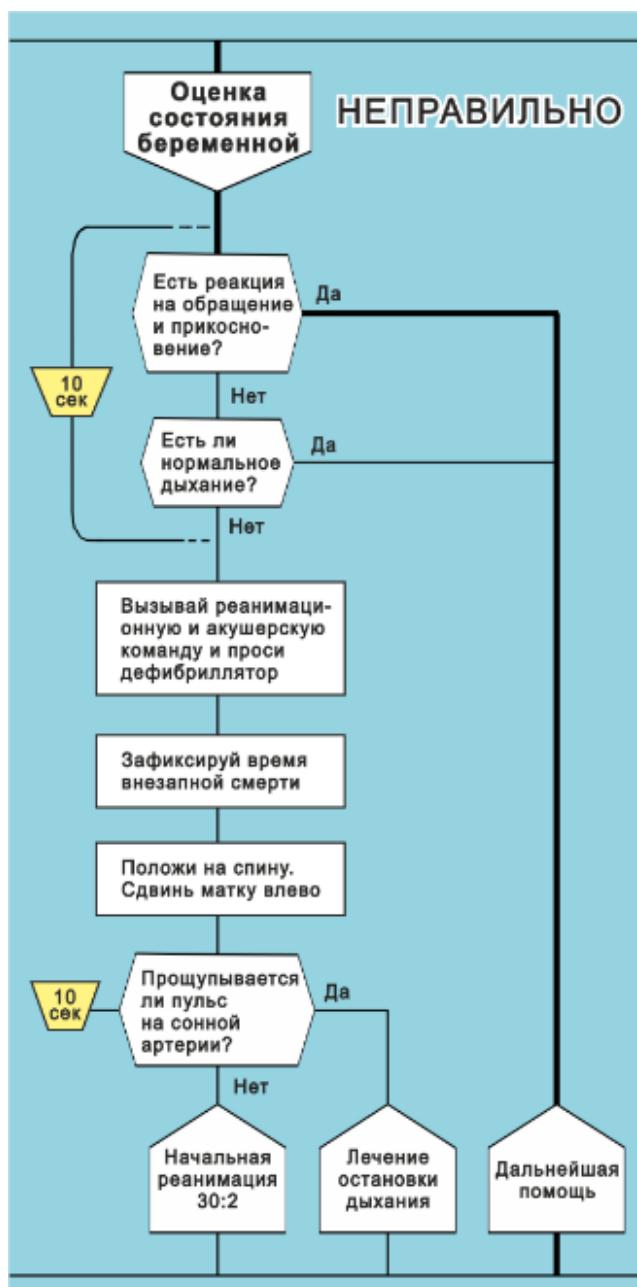


Рис. 107. Ветка с ошибкой. Нарушено правило «Чем правее, тем хуже»



Рис. 108. Ошибка исправлена с помощью рокировки

- «Начальная реанимация 30:2».
- «Лечение остановки дыхания».
- «Дальнейшая помощь».

Это значит, что нарушено основополагающее правило «Чем правее, тем хуже». Действительно, самая неблагоприятная для пациента ситуация «Начальная реанимация 30:2» по ошибке оказалась слева. А должна быть справа! Кроме того, икона Адрес «Дальнейшая помощь» должна занимать не правую, а крайнюю левую позицию.

Отсюда вытекает, что мы обнаружили эргономическую неточность (ошибку) в оригинале. Чтобы устранить дефект, необходимо несколько раз осуществить рокировку. Посмотрим, как это делается.

Первый раз выполняем рокировку в развилке «Есть реакция на обращение и прикосновение?». Это позволяет выпрямить главный маршрут и направить его по шампуру. Но этого мало.

Нужна вторая рокировка в развилке «Есть ли нормальное дыхание?». Благоприятный ответ Да мы также подключаем к шампуру.

Третью и последнюю рокировку делаем в развилке «Прослушивается ли пульс на сонной артерии?». Благодаря этому икона «Начальная реанимация 30:2» перемещается туда, куда надо — в крайнее правое положение.

В результате трех рокировок мы значительно улучшили схему. Но не полностью. Возникла неожиданная неприятность с верхней иконой «10 секунд». Если сохранить ее на прежнем месте (слева от ветки), нижний конец будет пересекать шампур. Это недопустимо, потому что пересечения категорически запрещены. Как же быть?

Выход есть! Надо передвинуть икону «10 секунд» и расположить ее справа от ветки, как показано на рис. 108. И пересечение сразу исчезнет. Такая операция называется *зеркальным отражением* иконы «Время группы» (отражение производится относительно вертикали).

Таким образом, мы решили все трудности. В результате трех рокировок и зеркального отражения удалось устраниć выявленные в первоисточнике нарушения правил языка ДРАКОН. Результат показан на рис. 108.

В заключение добавим, что два алгоритма на рис. 107 и 108 являются равносильными. Они отличаются лишь графически.

## ИЗУЧАЕМ ВЕТКУ «ЛЕЧЕНИЕ ОСТАНОВКИ ДЫХАНИЯ»

Ветка выделена из рис. 102 и показана на рис. 109. Как и ветка на рис. 105, она служит «большой развилкой» на три направления. И поэтому имеет три иконы Адрес.

Икона «2 минуты» прицеплена к иконе Действие «Вдувай 1 раз каждые 5–6 секунд». Смысл в том, что врач должен делать пациентке искусственные вдохи (вдувая в нее кислород каждые 5–6 секунд) в течение двух минут.

Нижние две иконы Время связаны с иконами Вопрос. Они указывают предельное время для выполнения операций. На принятие решения «Есть ли пульс?» врачу отводится 10 секунд. Чтобы выяснить «Есть ли дыхание?» дается 5–10 секунд.

Согласно *Правилу лишней иконы* перед иконами Вопрос отсутствуют иконы Действие с повторяющейся информацией.

Таким образом, икона Вопрос «Есть ли пульс?» имеет двойную функцию:

- Во-первых, врач реализует Действие «Нащупай пульс»,
- Во-первых, врач принимает Решение «Есть ли пульс?» с ответом Да или Нет.

Икона «Есть ли дыхание?» также реализует двойную функцию.

## ВЕТОЧНЫЙ ЦИКЛ

В главе 9 мы уже познакомились с *простым циклом*. Простой цикл легко узнать по характерной горизонтальной стрелке, направленной влево. Петля простого цикла всегда закручена против часовой стрелки (см. рис. 79, 81, 83).

Существует иной тип цикла, для которого стрелка не нужна. Он образуется с помощью иконы Адрес. Каким образом?

Если икона Адрес указывает на свою собственную (или на более левую) ветку, образуется *повторение действий*, то есть цикл.

Пример показан на рис. 109, где в иконе Адрес написано имя своей родной ветки «Лечение остановки дыхания».

Представим себе, что при выполнении этой ветки бегунок спускается вниз, попадает в икону Вопрос «Есть ли дыхание?», и врач видит, что дыхания нет. Это значит, что бегунок выходит через Нет, проходит через икону Адрес «Лечение остановки дыхания» и снова возвращается в начало той же самой ветки.

Если не изменятся условия в обеих иконах Вопрос, бегунок будет снова и снова утюжить ветку «Лечение остановки дыхания». Это и есть *веточный цикл*. Он обозначается двумя черными треугольниками (рис. 109).

Когда цикл закончит работу? Возможны два варианта:

1. Если у пациентки пропал пульс, бегунок покидает икону «Есть ли пульс?» через Нет и далее через икону Адрес прыгает в ветку «Начальная реанимация 30:2».
2. Если же, к счастью, у пациентки появилось дыхание, бегунок выходит из иконы «Есть ли дыхание?» через Да и затем через Адрес перемещается в ветку «Дальнейшая помощь».

Какова роль черных треугольников? Для работы алгоритма они совершенно не нужны. Они ни на что не влияют. Если их убрать, ничего не изменится.

Зачем же они поставлены? Только для того, чтобы служить удобным зрительным ориентиром.

Если бы треугольников не было, пришлось бы долго разбираться, чтобы убедиться в том, что в одной из икон Адрес записан адрес собственной ветки. А треугольники сразу бросаются в глаза.

Может случиться так, что в алгоритме потребуются несколько веточных циклов. В таком случае, чтобы не запутаться, можно использовать разные маркеры-треугольники, представленные в справочнике на рис. 22.

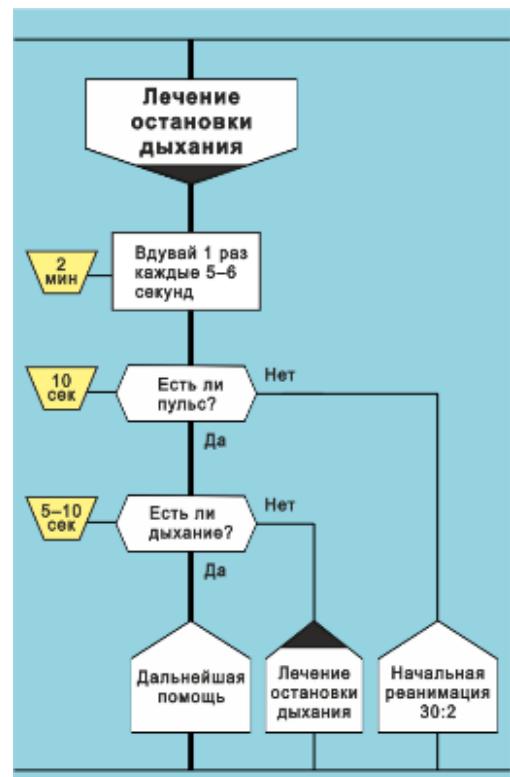


Рис. 109. Ветка «Лечение остановки дыхания»

Что такое  
веточный цикл

- Это цикл, при котором в иконе Адрес написано имя своей собственной (или более левой) ветки.
- Чтобы легко найти веточный цикл, его помечают двумя черными треугольниками.

## ИЗУЧАЕМ ВЕТКУ «НАЧАЛЬНАЯ РЕАНИМАЦИЯ 30:2»

В эту ветку можно попасть двумя способами:

- Из ветки «Оценка состояния беременной», когда в иконе «Прощупывается ли пульс на сонной артерии?» получен ответ Нет.
- Из ветки «Лечение остановки дыхания», когда из иконы «Есть ли пульс?» выходим через Нет.

Формула 30:2 означает, что спасатель поочередно выполняет:

- 30 компрессий грудной клетки,
- 2 вдоха в легкие пациентки.

На рис. 110 регулярное повторение этих действий изображено в виде цикла. Цикл заканчивается, когда принесут дефибриллятор.

Описание ветки «Применение дефибриллятора» мы опускаем, так как все используемые в ней иконы нам уже известны.

## ГЛАВНЫЙ МАРШРУТ СИЛУЭТА И ПРАВИЛО ВЕЗЕНИЯ

Предположим, что пациент является «везунчиком», в жизни ему всегда везет. Можно сказать, что жизнь такого пациента подчиняется «Правилу везения».

Заодно вспомним, что главный маршрут алгоритма — это наиболее благоприятный для пациента путь, ведущий из Заголовка в Конец. Слова «наиболее благоприятный маршрут» и «Правило везения» имеют один и тот же смысл.

Поясним сказанное применительно к алгоритму на рис. 102.

Чтобы построить главный маршрут силуэта, следует держаться Правила везения — во всех развилках надо выбирать вариант, который является наилучшим для нашей пациентки, беременной женщины.

Итак, приступим. Из Заголовка на рис. 102 путь идет вниз по шампуру первой ветки. Через икону Адрес бегунок попадает на шампур второй ветки. Здесь появляется вопрос: «Есть ли реакция на обращение и прикосновение?». Руководствуясь Правилом везения, бегунок отбрасывает плохой ответ и выбирает хороший, согласно которому пациентка успешно реагирует на прикосновение и обращение. Бегунок скользит вниз по шампуру второй ветки и опускается в икону Адрес «Дальнейшая помощь».

Таким образом, бегунок пропускает три ветки на рис. 102, описывающие экстренную помощь:

- «Лечение остановки дыхания».
- «Начальная реанимация 30:2».
- «Применение дефибриллятора».

В результате такого маневра бегунок обходит неприятности и сразу попадает на вход ветки «Дальнейшая помощь» (рис. 111). Здесь возникает второй и последний вопрос: «Нужна ли реанимация?»



Рис. 110. Ветка  
«Начальная  
реанимация 30:2»

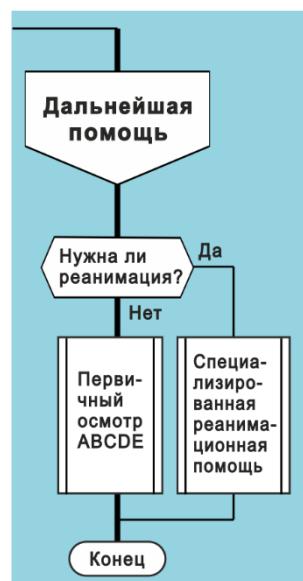


Рис. 111. Ветка  
«Дальнейшая помощь»

Правило везения требует выбрать ответ Нет. И бегунок по шампуру прямиком спускается до Конца.

Мы убедились, что главный маршрут силуэта на рис. 102 проходит через три шампера (через первую, вторую и последнюю ветки). Это и есть наиболее благоприятный маршрут для нашей беременной женщины.

## НЕЯСНОСТЬ НЕОБХОДИМО УСТРАНИТЬ

Внимательный анализ показывает, что главный маршрут алгоритма на рис. 102 содержит неясные места и порождает вопросы. Алгоритм посвящен реанимации, но главный маршрут по какой-то причине обходит все реанимационные действия.

В самом деле, можно ли, исследуя главный маршрут, говорить о том, что произошла внезапная смерть пациентки? Нет, нельзя. Можно ли считать, что у нее пропал пульс? Тоже нет. Наконец, были ли у нее трудности с дыханием? Опять-таки нет. Она нормально реагировала на обращение и прикосновение.

Главный маршрут алгоритма не содержит нужных сведений и требует обдумывания. По какой причине у врача возникли подозрения, что наша героиня попала в критическую ситуацию и, возможно, нуждается в реанимации?

Главный маршрут, к сожалению, не позволяет уточнить ситуацию и дать однозначный ответ. Приходится констатировать, что в данном алгоритме имеет место неясность. Неясность, которую необходимо устраниить.

Читатель может рассматривать описанную ситуацию как упражнение и по своему усмотрению устраниить недостаток.

## АЛГОРИТМ ВЫСОКОЙ ТОЧНОСТИ И НОВАЯ КУЛЬТУРА КЛИНИЧЕСКОГО МЫШЛЕНИЯ

ДРАКОН — алгоритмический язык высокой точности. Он представляет собой принципиально новый инструмент. Раньше таких инструментов просто не существовало.

ДРАКОН обеспечивает высочайшую точность при разработке медицинских алгоритмов. Это обнажает ряд проблем, которые раньше казались незаметными.

В медицине существуют определенные традиции, устоявшиеся привычки, традиционный стиль при описании медицинских алгоритмов. Для этого стиля характерен творческий, неформальный подход. Творческому стилю присуща некоторая размашистость, приблизительность, неточность и необязательность.

Творческий человек сосредоточен на главном, и эту главную идею он тщательно шлифует, совершенствует, уточняет, доводит до блеска и до победного конца.

Однако от главной идеи обычно ответвляются многочисленные боковые маршруты, за которыми очень трудно уследить.

Здесь требуются совсем другие качества: пунктуальность, дотошность, аккуратность, алгоритмическое буквоечество и алгоритмический педантизм.

Проблема состоит в том, что до сих пор медицина не знала требования о необходимости **полного анализа всех без исключения маршрутов алгоритма**.

Данное требование является принципиально новым. Оно знаменует переход к **новой культуре клинического мышления**.

## **СВЕРТКА ИНФОРМАЦИИ ИЛИ ВЫСОКАЯ ТОЧНОСТЬ?**

Существуют две точки зрения на медицинский алгоритм:

- алгоритм как свертка информации,
- алгоритм высокой точности.

Понимание алгоритма как свертки информации получило широкое распространение в медицине. По мнению профессоров Александра Краснова и Игоря Давыдкина, медицинский алгоритм предназначен для сворачивания информации. Они пишут:

«За последние годы вышло немало учебных пособий, где предприняты попытки написания профессиональных [медицинских] алгоритмов. Однако чаще всего эти алгоритмы написаны клиницистами в произвольном стиле, не имеют унифицированной общепринятой символики, перегружены числом элементов (счет которых иногда идет на десятки на странице. Поэтому они в принципе не могут быть восприняты как свертка информации (для чего, собственно, и предназначались)» [162] [163].

Так ли это? Можно ли согласиться с тем, что алгоритмы предназначались для свертки информации?

Свертка (сворачивание) информации означает сжатый комплекс информационных сведений, из которого удалено все лишнее и малозначительное, оставлена только самая суть. Излишнее сокращение делает информацию непонятной, при недостаточном сокращении за мелочами теряется суть вопроса [164].

По нашему мнению, трактовка алгоритма как свертки устарела и не имеет будущего. Она не позволяет осуществлять детальный анализ всех без исключения маршрутов алгоритма и является источником ошибок. На смену этой концепции приходит принципиально новая идея — идея медицинского алгоритма высокой точности.

## **ЭРГОНОМИЧНЫЙ АЛГОРИТМ**

Эргonomичный алгоритм — алгоритм, удовлетворяющий критерию сверхвысокого понимания. Преимущество эргономичных алгоритмов в том, что они намного понятнее, яснее, нагляднее и доходчивее, чем обычные.

Если алгоритм непонятный, в нем трудно или даже невозможно заметить затаившуюся ошибку.

И наоборот, чем понятнее алгоритм, тем легче найти дефект. Поэтому более понятный, эргономичный алгоритм намного лучше обычного. Лучше в том смысле, что он облегчает выявление ошибок, а это очень важно.

Ведь чем больше ошибок удастся обнаружить при проверке за столом, тем больше вероятность, что вновь созданный алгоритм окажется правильным, безошибочным, надежным. Кроме того, эргономичные алгоритмы удобны для изучения, их проще объяснить другому человеку.

Все в алгоритме понятно и ясно,  
Если он сделан эргономично.  
Эргономично – это прекрасно!  
Эргономично – значит отлично!

## КНИЖНЫЙ РАЗВОРОТ

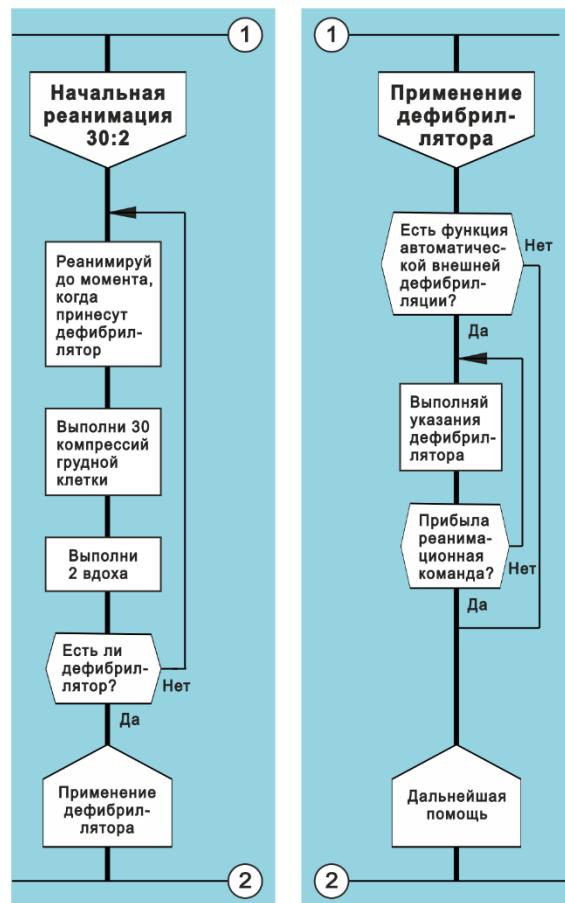
При изложении материала мы постепенно увеличивали сложность и объем алгоритмов. Самые крупные примеры представлены на рис. 96 и 102. Каждый из них занимает две страницы, образующие книжный разворот.

Возникает вопрос: как изображать большие алгоритмы в медицинских учебниках? Можно дать следующие рекомендации.

- Большие алгоритмы лучше размещать в виде силуэта на книжных разворотах.
- Если одного разворота мало, можно продолжить алгоритм-силуэт вправо по горизонтали и занять два (или даже три) соседних разворота.
- Сочленение двух разворотов осуществляется с помощью двух икон Соединитель, как показано на рис. 112 (по два соединителя с каждой стороны).
- Для книги, содержащей медицинские алгоритмы, желательно использовать формат 70×100 1/16.
- Не следует использовать слишком мелкий шрифт внутри икон; кегль (размер буквы по вертикали) должен быть не меньше 6 или 7 типографских пунктов.

## ВЫВОДЫ

1. Многоадресный силуэт — это силуэт, содержащий многоадресные ветки.
2. Многоадресный силуэт позволяет:
  - изменять последовательность выполнения веток,
  - выборочно запускать ветки, пропуская некоторые из них,
  - создавать веточные циклы.
3. В языке ДРАКОН используются эффективные средства управления восприятием:
  - картографический принцип силуэта,
  - фиксация начала и конца,
  - шапка.



**Рис. 112.** Переход с листа на лист производится с помощью двух икон Соединитель

- увеличение толщины шампуров,
  - выделение шрифтом и кеглем,
  - акцент на иконах Время,
  - треугольные маркеры,
  - ветка-комментарий.
4. Ветка-комментарий позволяет выделить из пояснительного текста наиболее важные пояснения и вставить их непосредственно в чертеж алгоритма.
  5. Иконы Время указывают врачу предельное время, отводимое на ту или иную операцию.
  6. Если две иконы (Действие и Вопрос) соединены последовательно, причем текст в обеих иконах почти полностью совпадает, икона Действие в медицинском алгоритме является лишней и подлежит удалению.
  7. Веточный цикл образуется в случае, если икона Адрес указывает на свою собственную или на более левую ветку.
  8. Понимание алгоритма как свертки информации устарело. Будущее принадлежит медицинским алгоритмам высокой точности.
  9. Алгоритмы высокой точности нацелены на решение принципиально новой задачи — описание всех без исключения маршрутов алгоритма.

## Глава 15

# МЕДИЦИНСКАЯ АЛГОРИТМИЧЕСКАЯ СИСТЕМА

## ОТ ОДНОЧНЫХ АЛГОРИТМОВ — К СИСТЕМЕ АЛГОРИТМОВ

До сих пор мы описывали *одиночные* алгоритмы. Начиная от простейших, таких как «Действия при травматической ампутации частей тела» (рис. 1) и «Помощь при небольшой ране» (рис. 25). До более сложных, таких как «Первая помощь при химическом ожоге глаз» (рис. 96) и «Реанимация беременной женщины» (рис. 102).

Постепенно мы переходили от одиночных алгоритмов, полностью изолированных друг от друга, к взаимосвязанным, решающим общую задачу.

Здесь следует указать на *принципиально новый* момент. В данном случае мы исключаем из рассмотрения любые словесные пояснения и аргументы. Если врач скажет: «Мамой клянусь, что между алгоритмами есть связь», это всего лишь пустословие.

Связь должна быть не словесной, а формальной, она должна иметь точное графическое обозначение. В языке ДРАКОН таким графическим символом служит икона Вставка.

При каком условии можно утверждать, что между двумя алгоритмами имеется связь? При условии, что в первом алгоритме есть икона Вставка, которая точно указывает на второй алгоритм. При отсутствии Вставки говорить о наличии связи бессмысленно и недопустимо. Вставка играет роль клея, скрепляющего алгоритмы между собой. Нет клея — нет и связи. Нет связи — нет и системы алгоритмов.

## ПРИМЕР

Первый опыт построения алгоритмической системы мы продемонстрировали, когда на сцене появился скромный дуэт: «Реанимационные действия при наличии у новорожденного кистозной гигромы» (рис. 23) и «Использование пульсоксиметра» (рис. 24).

В первом алгоритме предусмотрена Вставка, внутри которой написано «Использование пульсоксиметра». Мы видим, что название Вставки и название второго алгоритма совпадают. Совпадение не случайно — оно означает, что Вставка однозначно указывает на второй алгоритм. В результате образуется связь между алгоритмами, и они превращаются в алгоритмическую систему.

Конечно, это очень простой и малоинтересный пример. Впрочем, лиха беда начало.

## ВТОРОЙ ПРИМЕР

На рисунке 83 показан алгоритм «Отсасывание мекония из трахеи новорожденного». В нем используются две иконы Вставка:

- «Интуибирование трахеи»,
- «Выполнение искусственной вентиляции легких».

Каждая из них представляет собой довольно сложный алгоритм. Раньше мы ограничивались замечанием, что указанные алгоритмы обязательно должны быть раскрыты где-нибудь в подходящем для этого месте. Причем читатель должен иметь возможность попасть в это место и ознакомиться с алгоритмом. Если такой возможности нет, значит, Вставка превращается в шараду: «Пойди туда, не знаю куда».

Замечание правильное, но суть не в этом. Суть в том, что икона Вставка — это фундаментальная основа для строительства алгоритмических систем.

## ТРЕТИЙ ПРИМЕР

Обратимся к рисунку 83, где представлен алгоритм «Реанимация беременной женщины». В нем используются две иконы Вставка:

- «Первичный осмотр ABCDE»,
- «Специализированная реанимационная помощь».

Последняя Вставка — сложнейший алгоритм. Специализированная реанимационная помощь — это целая отрасль медицины. Близкая по теме книга «Специализированная реанимация новорожденного, построенная на основе языка ДРАКОН, насчитывает 400 страниц [127].

На этом мы заканчиваем рассказ о трех примерах алгоритмической системы, которые удалось «втиснуть» в книгу. Ясно, что этого мало.

Ниже, в качестве компенсации, в общем виде и без иллюстраций рассмотрен большой пример на тему скорой помощи<sup>12</sup>.

---

<sup>12</sup> Чтобы раскрыть тему «Медицинская алгоритмическая система» и снабдить ее иллюстрациями, необходимо написать отдельную книгу. На этих страницах сделать это невозможно.

## СКОРАЯ МЕДИЦИНСКАЯ ПОМОЩЬ И БОЛЬШИЕ АЛГОРИТМЫ

Что такое скорая помощь? Можно ли представить ее как единый стройный алгоритм? Алгоритм, который делится на множество отделов, подразделов и более мелких подразделений?

Мы сразу же сталкиваемся с парадоксом. С одной стороны, практика скорой помощи хорошо организована, тысячу раз проверена и отработана до звона. Накоплен огромный опыт. Казалось бы, должны существовать и войти в учебники детальные графические алгоритмы скорой помощи. Однако этого нет. Все попытки отыскать в литературе подробные чертежи «большого алгоритма» скорой помощи, учитывающие все без исключения особенности ее работы, ни к чему не приводят. Ничего подобного нет даже близко<sup>13</sup>.

Почему? Читатель уже знает наш ответ. Потому что отсутствует удобная и строгая графическая нотация. Без удобной нотации, как без рук — ничего не выйдет. Медицинский язык ДРАКОН создан для того, чтобы коренным образом исправить положение и предложить эффективное решение.

В данной главе мы ведем речь об использовании ДРАКОНа для разработки больших, если угодно, сверхбольших и даже громадных алгоритмов. В качестве важного и сложного примера мы выбрали скорую помощь. Однако наши рассуждения полностью распространяются на любые большие медицинские алгоритмы.

## АЛГОРИТМИЧЕСКАЯ СИСТЕМА

*Медицинская алгоритмическая система* — специальный термин, обозначающий не один алгоритм, а совокупность логически связанных графических алгоритмов, решающих общую задачу и направленных на исчерпывающее процедурное описание медицинской темы, например, скорой помощи. Это не одиночный частный алгоритм, а именно совокупность алгоритмов, т. е. система.

Алгоритмическая система имеет определенную структуру. Она включает головной алгоритм (алгоритм верхнего уровня) и алгоритмы более низких уровней (рис. 112а).

Головной алгоритм содержит несколько икон Вставка. Например, десять вставок. Обозначим их через A, B, C ... K. Каждой вставке соответствует одноименный алгоритм низкого уровня. Вставке A соответствует алгоритм A. Вставке B соответствует алгоритм B. И так далее до K (рис. 112а).

Предположим, что алгоритм A, в свою очередь, разбивается на десять алгоритмов третьего уровня: A1, A2, ... A10. Точно так же дробятся и остальные алгоритмы B, C, ... K. Подытожив, получим, что алгоритмическая система, изображенная на рис. 112а, состоит из 111 алгоритмов (цифры условные).

## ИЕРАРХИЧЕСКАЯ СИСТЕМА АЛГОРИТМОВ

На рис. 112а представлена иерархическая система алгоритмов, имеющая три уровня иерархии. На верхнем уровне один алгоритм, который является головным. На втором уровне живут 10 алгоритмов, подчиненных головному (A, B, C, ... K). На третьем, самом нижнем уровне иерархии находятся 100 алгоритмов, на которые указывают указательным пальцем Вставки второго уровня.

<sup>13</sup> Сокращенные алгоритмы, такие как [79], являются, хотя и полезными для определенных случаев, но в целом неэргономичными и неудовлетворительными.

Уровней иерархии может быть и больше. Все зависит от конкретного случая.

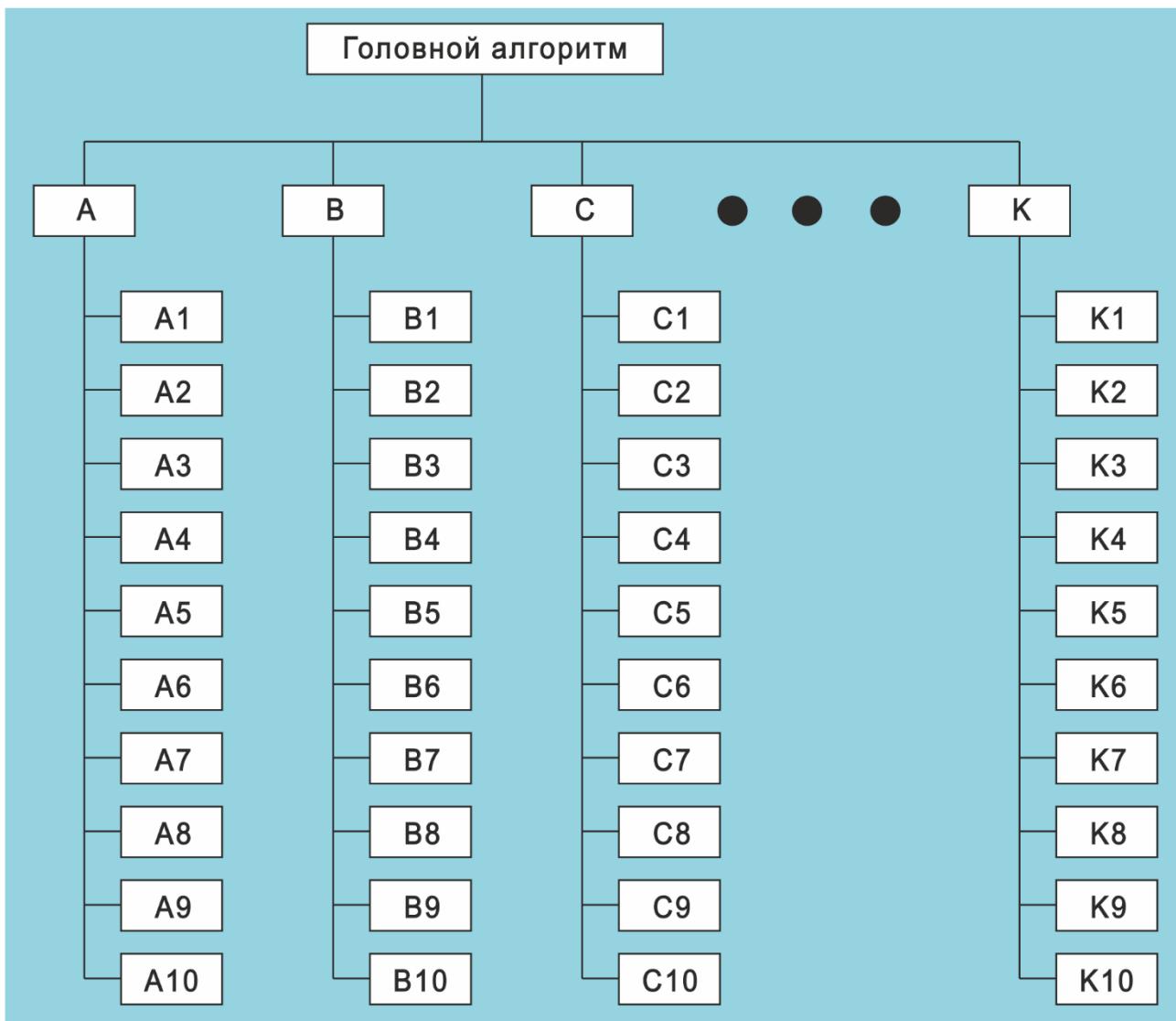


Рис. 112а. Алгоритмическая система, состоящая из 111 алгоритмов

## РАЗДЕЛЯЙ И ВЛАСТВУЙ!

Принцип «Разделяй и властвуй» означает, что головной алгоритм делится на несколько алгоритмов второго уровня. Их может быть семь, десять или двадцать с лишним. В свою очередь каждый алгоритм второго уровня делится на алгоритмы третьего уровня. И так далее.

Что же мы при этом выигрываем?

С помощью принципа «Разделяй и властвуй» получаем огромный выигрыш. Первоначально проблема скорой помощи представляет собой некую гигантскую аморфную массу, похожую на первобытный хаос. И никто не знает, как описать ее в виде удобного графического алгоритма. С какого боку к ней подступиться.

С чего начать? Можно начать сверху. Прежде всего, нужно выделить головной алгоритм. Это должен быть силуэт. Не надо увлекаться гигантоманией и строить силуэт с большим числом веток. Желательно, чтобы головной силуэт был содержал

не более 10–20 веток, лучше еще меньше. Примерно так же следует проектировать алгоритмы второго и нижних ярусов.

Можно начать наоборот, не сверху, а снизу и постепенно продвигаться снизу вверх. Можно сочетать оба метода.

В конечном итоге получится четкий и взаимоувязанный большой алгоритм, аккуратно нарезанный на мелкие и удобные порции. Это выгодный прием, позволяющий строить алгоритмы по принципу «Разделяй и властвуй», т. е. расчленять алгоритмы на части и снова собирать их из частей (вставок).

Нисходящий процесс можно назвать *декомпозицией* (разбиением на части) большого алгоритма, например, алгоритма скорой помощи. В языке ДРАКОН для декомпозиции больших алгоритмов предусмотрены два инструмента:

- использование Вставок,
- дробление алгоритмов на ветки.

## ВОЗРАЖЕНИЯ СКЕПТИКОВ

— А можно ли это сделать? Удастся ли довести процесс до конца?

— Чепуха! Ничего у вас не выйдет! Типичная маниловщина.

— Вздорная утопия!

— Пустая трата времени! Никому это не нужно!

Чтобы ответить на обвинения, заручимся поддержкой пионера медицинской алгоритмизации профессора Владимира Тавровского.

Ниже приводятся выдержки из его работы «Алгоритмы действий врача скорой помощи» [165]. Выдержки пронумерованы и оформлены как эссе (всего 20 эссе), некоторые снабжены комментарием.

### Эссе 1.

#### АЛЛО! ГОВОРИТ ПРОФЕССОР ТАВРОВСКИЙ

— 1980 году Юрий Янкин, зам. Главврача Новокузнецкой станции скорой помощи предложил создать алгоритм для «скоростников» — врачей его станции. Вместе мы и придумали такой алгоритм. Он предназначался для кардиологов и назывался «Сердце, сосуды». Как он выглядел? Это было дерево решений, изложенное на перфокартах, скреплённых в маленький альбом.

Алгоритм выдавался кардиологическим и реанимационным бригадам, причём пользование им было обязательным.

Затем были созданы алгоритмы и по другим разделам скорой помощи. В их составлении, совершенствовании и эксплуатации приняли участие станции Новосибирска, Барнаула и Перми, а затем — Красноярска и Улан-Удэ [165].

## КОММЕНТАРИЙ 1

Перфокарты, на которых изображен текст алгоритма, канули в прошлое. Навсегда. Здесь важно другое — Тавровский описывает первую попытку применить алгоритмический подход в работе станций скорой помощи. Попытка была успешной. На примере кардиологических и реанимационных бригад показана принципиальная возможность алгоритмизации скорой помощи.

О каких перфокартах идет речь? Тавровский имеет в виду перфокарты с двухрядной краевой перфорацией формата К5; на карте напечатан алгоритм, состоящий из дерева решений и врачебных назначений [166].

## Эссе 2. МНОГОЕ УСТАРЕЛО, НО ОСТАЛИСЬ МЕТОДИКА И ПРИНЦИПЫ

Помимо диагностических и лечебных мероприятий, алгоритмы регламентировали показания к вызову другой бригады и транспортировке. Они подсказывали:

- куда именно надо транспортировать пациента,
- когда обратиться за советом к старшему врачу смены,
- когда запланировать активный повторный вызов к больному, оставленному дома и т.п.

Я наблюдал за использованием этих средств несколько лет, но сейчас об их судьбе ничего не знаю. Несомненно, их содержание частично устарело. Однако, даже если многое отстало от жизни, остаётся методика их составления и принципы использования, проверенные обширной практикой. А это ведь и есть ответ на вопросы «Зачем это нужно?» и «Как это сделать?» [165].

## КОММЕНТАРИЙ 2

Принципы алгоритмизации Тавровского имеют непреходящую ценность. Тавровский был первопроходцем и работал в труднейших условиях, не имея удобной графической нотации для записи алгоритмов. Тем не менее, он добился замечательных успехов.

## Эссе 3. КАК РАСПИСАТЬ НАПЕРЁД ДЕЙСТВИЯ ВРАЧА ВО ВСЕХ ВОЗМОЖНЫХ СИТУАЦИЯХ (на примере алгоритма «Сердце, сосуды»)

Врач скорой помощи прибыл к больному и решил, что это — сердечный больной.

- Какие вопросы врач должен задать себе в первую очередь?
- В каком порядке?
- Какие ответы он может получить?
- К чему будет обязывать каждый из вариантов ответа?
- Что он должен будет сделать?
- Сколько времени можно ждать эффекта от принятых мер?
- Что делать, если меры не дали результата?

Перечисленные вопросы и ответы на них (все возможные вопросы и все вероятные ответы) надо изложить в виде заранее продуманной схемы алгоритма. В виде дерева решений.

Затем, в реальной ситуации следует пользоваться этой схемой как руководством к действию. Схема должна быть не только всеобъемлющей, детальной и точной, но и компактной, удобной для пользования в любой обстановке.

Лучший способ доказать, что такая схема возможна — предъявить её. Но как это сделать в рамках интернет-рассылки? Приходится полагаться на отдельные примеры и уповать на доверие и воображение читателей [165].

## КОММЕНТАРИЙ 3

Тавровский разумно ставит вопросы, на которые должен дать ответ алгоритм. Он говорит: «Лучший способ доказать, что такая схема возможна — предъявить её». К сожалению, схему алгоритма он не предъявляет. Почему? Потому что у него не было удобной и строгой графической нотации для записи алгоритмов.

Язык ДРАКОН позволяет устранить этот пробел, т. е. создать нужную схему и распространять ее как в бумажном виде, так и в сети Интернет. Поэтому «уповать на доверие и воображение читателей» совершенно не нужно.

Имея комплект дракон-схем, читатели могут читать алгоритм своими глазами, а не с помощью воображения.

## Эссе 4.

### ЖИВ ЛИ БОЛЬНОЙ? НУЖНА ЛИ РЕАНИМАЦИЯ?

Первая задача кардиолога — решить, жив ли больной.

Соответственно, первый вопрос алгоритма: «Сознание?» Возможны ответы:

- Сохранено.
- Спутанное.
- Отсутствует.

Первые два ответа предполагают сохранение дыхания и кровообращения, при третьем надо спросить: «Дыхание?». И подсказать три ответа:

- Сохранено, эффективное.
- Сохранено, неэффективное.
- Отсутствует.

Последние два ответа означают необходимость реанимации. Какой именно реанимации? При отсутствии дыхания для выбора адекватных реанимационных мер надо спросить: «Сердечная деятельность?». Возможны три ответа:

- Сохранена.
- Фибрилляция.
- Асистолия.

Таким образом, тремя вопросами («Сознание?», «Дыхание?», «Сердечная деятельность?») определяется, необходима ли реанимация и какая именно.

Далее, в каждом из вариантов, при которых требуется реанимация, будут предписаны все меры, необходимые при этом варианте, и только они. В каждом будет указано, по каким критериям, по каким результатам предпринятых действий надо выбрать следующий шаг [165].

## Эссе 5.

### ЧТО ДЕЛАТЬ, ЕСЛИ РЕАНИМАЦИЯ НЕ НУЖНА?

Если реанимация не нужна, надо решить, какое состояние является определяющим. Таких состояний может быть только пять:

- Резко сниженное артериальное давление.
- Резко повышенное артериальное давление.
- Нарушение сердечного ритма.
- Болевой синдром.
- Одышка.

Простыми вопросами о болевых ощущениях, одышке, давлении и пульсе можно сделать не только нужный выбор, но и распознать комбинацию этих состояний. Например, боль, сочетанную с коллапсом, аритмию с одышкой и т. д.

Разумеется, для получения ответов надо предложить:

- измерение давления,
- выслушивание сердца и лёгких,
- подсчёт пульса и дыхания,
- электрокардиограмму.

С помощью этих средств ответы будут получены. И станет ясно, какое срочное лечение необходимо. Далее можно к этому лечению приступить [165].

## Эссе 6. ШАГ ЗА ШАГОМ

Указанная информация, в виде дерева вопросов и ответов, умещается на двух пронумерованных карточках размером в половину стандартной страницы. Дерево разветвляется и ведет к трём десяткам точек, в каждой из которых надо действовать по-своему. Причем для каждой точки ясно, как надо действовать.

Пронумеруем точки. Для каждого номера составим свою карту необходимых мероприятий. Отправим врача с каждой такой точки на соответствующую ей карту. На карте написано, что именно надо сделать (медикаменты, процедуры, контрольные измерения) и сколько минут ждать результата. Это лечебные назначения.

Когда они будут выполнены, снова должен последовать вопрос о состоянии пациента (сознание, боль, давление, одышка, пульс) [165].

## Эссе 7. РЕЗУЛЬТАТ МОЖЕТ БЫТЬ РАЗНЫМ

Укажем три варианта.

- Во-первых, состояние может перейти в другое грозное сочетание тех же симптомов. Для каждого из них у нас уже есть карта. Отшлём врача туда. Для этого укажем под результатом номер очередной карты, необходимые в данный момент действия изложены там.
- Во-вторых, состояние может остаться прежним. Тогда надо задать вопрос о суммарной длительности мероприятий и предложить в зависимости от неё:
  - либо повторение тех же мер,
  - либо их усиление,
  - либо транспортировку в больницу.
- В-третьих, состояние может нормализоваться. Тогда следует задать новые вопросы, чтобы сделать выбор:
  - оставить больного дома

— или везти в стационар [165].

## **Эссе 8. КАК ТРАНСПОРТИРОВАТЬ БОЛЬНОГО**

Транспортировка — специфическая область решений врача скорой помощи. Здесь не всё просто. Надо убедиться, что пациент транспортабелен и что его может транспортировать именно эта бригада. В противном случае следует вызвать другую бригаду.

Больной может отказаться от транспортировки. Тогда надо дать ему чёткие рекомендации, а старшему дежурному врачу сообщить, что через некоторое время к пациенту нужен активный выезд.

Надо предусмотреть специальные лечебные мероприятия во время транспортировки. Наконец, надо правильно выбрать больницу [165].

## **Эссе 9. КОГО КУДА?**

Выбор больницы зависит от ряда сугубо местных причин. Больных, перенесших клиническую смерть, обычно везут в соответствующий реанимационный центр, больных с инфарктом — в инфарктное отделение.

Когда ситуация менее грозная, выбор усложняется. Играют роль район города, возраст больного, иногда его профессия или профессия члена семьи. Так например, в Новокузнецке всех работников Кузнецкого металлургического комбината и членов их семей госпитализировали только в медсанчасть комбината, а других — в зависимости от места жительства.

Всё это надо решать быстро. При таком количестве факторов немудрено и ошибиться. Ошибка приведёт к отказу в госпитализации и перетранспортировке в другую больницу.

Поэтому транспортировке посвящалась отдельная карта, которая учитывала местные условия. Данная карта существовала в разных вариантах: новосибирском, новокузнецком, барнаульском и пермском. Пользуясь картами, врачи действовали безошибочно: отказы в госпитализации и перетранспортировки исчезли вообще [165].

## **Эссе 10. ПОЧЕМУ ВРАЧ НЕИЗБЕЖНО ОГРУБЛЯЕТ РЕШЕНИЯ?**

Выше приведено, конечно, слегка упрощённое описание алгоритма. Но только слегка. Основная канва выглядит именно так. Читателю придётся поверить, что все многообразие тактики кардиолога на этапе скорой помощи умещается в полусотне карточек (правда, с обеих сторон). Многообразие, как видите, не безграничное, доступное компактному изложению.

Очевидно, держать 50 вариантов в уме не так-то просто. Кроме того, нужно быстро и точно выбирать между ними. Это тоже не просто.

Но и это не все. Ведь элементарные ситуации по-разному сочетаются, по-разному переходят друг в друга, повторяются. Тут уж никому не под силу находить наилучшие пути по памяти.

К чему это приводит? К тому, что в обычной практике врач неизбежно упрощает, огрубляет свои решения. Он пользуется меньшим числом стереотипов, чем то, которое выделяют специалисты при спокойном рассуждении в кабинетной тиши.

Алгоритм как раз и позволяет устраниить недостаток. Он дает возможность сохранить тщательно дифференциированную, рассудительную, отточенную тактику, созданную заранее, в спокойной обстановке. И передать нужные решения врачу прямо к постели больного [165].

## **Эссе 11. КАК ПОЛЬЗОВАТЬСЯ АЛГОРИТМОМ**

Алгоритм разделен на части, построен по модульному принципу. Каждая клиническая ситуация размещена на отдельной карте. Работа всегда начинается с первой карты, дальше алгоритм ведёт врача сам. Начало каждой карты, кроме первой — результат решения, принятого на предыдущей карте.

Решения, которые будут приняты здесь, уведут на другие карты либо будут означать завершение работы.

Текст карты — это вопросы, ответы и рекомендации о действиях. После совершения действий следует очередной вопрос. Врач пробегает глазами схему, переходит на очередную карту, получает рекомендации, выполняет их, получает результат. И затем готов ответить на следующий вопрос.

В карте обслуживания вызова делаются обычные текстовые записи о состоянии больного и принятых мерах. Кроме того, на левом поле карты записываются столбиком номера пройденных карт. Столбик из 3-5 номеров (может быть и больше) — это и есть протокол работы по алгоритму. По нему всегда можно точно восстановить ход рассуждений врача.

Конечно, врач вправе поступить «не по алгоритму», отклониться. Но при отклонениях он должен указать, чем они вызваны и в чём состоят [165].

## **Эссе 12. КАКИЕ АЛГОРИТМЫ ВЫДАВАЛИСЬ БРИГАДАМ СКОРОЙ ПОМОЩИ**

Следует обратить внимание на сугубо технические вещи. Карты должны быть напечатаны чётким и крупным шрифтом. Соединить их в альбом надо так, чтобы он легко раскрывался, как перекидной календарь.

Перед самыми важными и часто употребляемыми картами надо закрепить закладки, чтобы алгоритм легче всего раскрывался именно здесь. Вот и всё. Именно такие альбомы были в составе снаряжения бригад скорой помощи в Новокузнецке и других городах Сибири. Именно так они использовались [165].

## **Эссе 13. ПОЧЕМУ ВСЕ БРЮЗЖАТ И ВСТРЕЧАЮТ В ШТЫКИ?**

Когда в устоявшийся порядок со всеми его недостатками, по поводу которых все брюзжат, вносится что-то новое, первая реакция большинства — не пушть! Не взглянуться, не подумать, а отторгнуть и для этого высмеять или сочинить страшилки.

Чего только не наслушался я по поводу алгоритмов:

- Нельзя ломать стереотипы врача!
- Врач будет лечить шаблонно, без учёта индивидуальности больного!
- Врач отучится думать сам!
- Нельзя больного рассматривать через дырку в перфокарте!
- А кто будет отвечать в случае чего?
- Врач, копающийся в алгоритме, вызовет недоверие у больного!
- Надо срочно помогать, а вместо этого врач будет схемки листать! [165].

## **Эссе 14.** **ЧУШЬ ВЕДЬ ВСЁ, А КАК ЗВУЧИТ!** **КАК ВОЗБУЖДАЕТ!**

Приходилось и приходится отвечать на упреки. Начнём с последнего. Конечно же, врач, получив первое представление о состоянии пациента, начинает действовать: отдаёт распоряжения помощникам, и, если нужно, сам включается в действия.

Но наступает момент, когда можно перевести дух и сделать записи в карте вызова. В этот момент врач и обращается к алгоритму. Алгоритм подскажет, что было упущено и каким должен быть следующий шаг. И каким — в зависимости от того, что получится — может быть следующий.

Алгоритмом всего-навсего оснащается процесс обдумывания и протоколирования, который есть всегда. На который всегда тратится некоторое время [165].

## **Эссе 15.** **ИМЯ АЛГОРИТМ, ВРАЧИ СТАЛИ** **ДЕЙСТВОВАТЬ УВЕРЕННЕЕ**

Не может толковая программа действий отучить врача думать. Наоборот, она приучает его думать логично, обогащает набор его стереотипов.

Не может пошаговый алгоритм привести к шаблонному лечению. Именно потому, что он пошаговый, тактика врача развивается в соответствии с реальным ходом событий. И, конечно, всегда за всё отвечает врач, потому что алгоритм, как всякая книга, как всякое пособие, только подсказывает решение. А принимает его человек, имеющий право не послушаться совета.

Любопытно, что врачи скорой помощи принимали новшество с интересом. Очень скоро обнаружилось, что с алгоритмом они стали действовать и дифференцированнее, и увереннее. Это объясняется просто: в обстановке экстренности хорошо иметь перед глазами перечень необходимых действий.

Кроме того, врач получает возможность чётко аргументировать свои действия в обоих случаях: и тогда, когда он следует алгоритму, и когда отклоняется от него [165].

## **Эссе 16.** **О ДРУГИХ АЛГОРИТМАХ И НЕКОТОРЫХ ЭФФЕКТАХ**

Из алгоритма «Сердце, сосуды» для специализированных бригад был легко получен алгоритм «Линейная бригада — Сердце, сосуды». Для этого оказалось достаточным те действия, которые может осуществить только кардиолог, заменить

более простыми. И дополнить рекомендацией о вызове кардиологической бригады «на себя».

По тем же правилам был составлен алгоритм «Дети» для детских реаниматоров, а затем — его аналог «Линейная бригада — Дети».

Следом появились «Дыхание. Живот. Урология. Гинекология» для линейных бригад, а также «Травмы. Шок. Коллапс» и «Отравления. Интоксикации» — для бригад реанимационных. Все они применялись так же, как было описано. Все принесли положительные результаты [165].

## **Эссе 17. ПРЕИМУЩЕСТВА**

Использование алгоритмов скорой помощи позволило получить ряд преимуществ:

- более дифференциированную тактику,
- более частое и обоснованное применение специальных средств и медикаментов,
- исключение случаев неправильной транспортировки и отказов в госпитализации,
- практическое исчезновение жалоб на скорую помощь (это связано с уменьшением срока ожидания обслуживания),
- уменьшение числа повторных вызовов.

Старший дежурный врач, не будучи специалистом во всех разделах, смог с помощью алгоритмов проверять по свежим следам правильность и полноту действий врача на вызове [165].

## **Эссе 18. АЛГОРИТМЫ БЫСТРО ВПИТАЛИ КОЛЛЕКТИВНЫЙ ВРАЧЕБНЫЙ ОПЫТ**

Предметом особого внимание старших врачей и старших специалистов станции стали случаи отклонений от алгоритма. За ними всегда стоит либо неразумное своеволие врача, либо его недостаточная оснащённость, либо упущения в самом алгоритме. Последние врач справедливо компенсировал своей инициативой и своим опытом.

Каждый из этих вариантов позволяет предпринять действия, которые будут совершенствовать работу. Либо врач признает и намотает на ус свою ошибку, либо алгоритм будет дополнен и усовершенствован.

Чёткое изложение врачебной тактики привлекло внимание врачей, особенно активных и знающих, побудило их внести свою лепту, указать на возможности совершенствования тактики, на использование новейших методов. Благодаря этому, алгоритмы быстро впитали коллективный врачебный опыт и поставили его на службу всем и каждому [165].

## **Эссе 19. ПОЧЕМУ НУЖЕН НОУТБУК**

Можно ли алгоритмы автоматизировать? Сегодня можно, если обеспечить выездную бригаду не альбомом, а ноутбуком. Запрограммировать то, что уже

изложено в виде алгоритма, не сложно. Инструментарий для этого существует, я его приготовил.

Но тогда надо сделать и следующие шаги: автоматически протоколировать ход рассуждений и действий врача, печатать этот протокол, переносить такие протоколы из ноутбука в центральную базу данных, наращивать и анализировать эту базу и т.д. Автоматизировать так автоматизировать. Всё это совершенно ясно можно себе представить. Возможность появилась потому, что сначала удалось чётко изложить рациональную врачебную деятельность в реальной обстановке [165].

## Эссе 20. ФОРМАЛИЗОВАТЬ ВРАЧЕБНУЮ МЫСЛЬ. СДЕЛАТЬ ЕЁ ЧЁТКОЙ, ЛОГИЧНОЙ, ГЛУБОКОЙ

Хотя алгоритмы для врачей стационара и поликлиники были сделаны и использованы тремя годами раньше, именно на станции скорой помощи особенно демонстративно выявились возможность формализовать врачебную мысль, сделать её более чёткой, логичной и глубокой.

Здесь же наиболее резко проявилась зависимость конечных результатов медицинской помощи от этой чёткости и глубины. Тем не менее, алгоритмизация врачебных решений вообще, везде, во всех областях — вещь не простая. Она требует не только удобной для врача реализации и организационных решений. Необходимо также серьёзное теоретическое осмысление [165].

## ОБСУЖДЕНИЕ

На этом мы заканчиваем излагать выступление Владимира Тавровского. Мы привели 20 эссе, представляющие собой выдержки и почти буквальные цитаты из его работы «Алгоритмы действий врача скорой помощи» [165].

Зачем мы это сделали? Чтобы на примере скорой помощи доказать возможность точного и полного описания больших медицинских алгоритмических систем.

Заслуга Тавровского в том, что он был первопроходцем, который сумел своим зорким глазом — глазом главного врача туберкулезной больницы — увидеть проблему алгоритмизации медицины во всей ее восхитительной многосложности. И предложить решение не только теоретическое, но и конкретное, практическое. Он обобщил свои взгляды на алгоритмизацию медицины в двух книгах:

- Лечебно-диагностический процесс: Теория. Алгоритмы. Автоматизация. 1997 год [69].
- Автоматизация лечебно-диагностического процесса. 2009 год [167].

Однако практическое решение Тавровского столкнулось с трудностями и прекрасная идея, поначалу набиравшая реальную силу в медицинских учреждениях Сибири и обещавшая заманчивые перспективы, постепенно сошла на нет и почти заглохла. И, к сожалению, не получила повсеместного распространения.

Почему? Это вопрос сложный, и мы не будем его касаться. Вспоминаются слова Пушкина:

Свободы сеятель пустынный,  
Я вышел рано, до звезды;  
Рукою чистой и безвинной  
В порабощенные бразды  
Бросал живительное семя —  
Но потерял я только время,

## ЧЕМ ОТЛИЧАЕТСЯ ЯЗЫК ДРАКОН ОТ КОНЦЕПЦИИ ТАВРОВСКОГО?

Прежде всего, укажем, что их объединяет. Обе идеи относятся к одной теме, которая называется «алгоритмизация медицины». Последнее понятие очень широкое и допускает различные толкования.

Тавровский трактует его в широком смысле и включает в сферу своих исследований и интересов три варианта внедрения алгоритмов в медицинскую практику:

- неавтоматизированный,
- частично автоматизированный,
- полностью компьютеризованный [168].

Мы же, наоборот, стремимся максимально ограничить сферу своих притязаний и с этой целью вводим нарочито узкий термин «алгоритмизация медицинской литературы».

Пересекается ли наша тема с идеями Тавровского? Правильнее сказать, что они дополняют друг друга.

Излагая свои идеи, Тавровский использует профессиональный медицинский язык. Он не критикует его и использует как данность, как нечто незыблемое и неизменное.

Мы занимаем противоположную позицию; для нас язык — объект критики и совершенствования. Вот названия наших статей:

- Можно ли улучшить медицинский язык? [169].
- Алгоритмизация медицины и реформа медицинского языка [170].

ДРАКОН не противоречит концепции Тавровского; наоборот, он может исправить ее слабые места и увеличить пробивную силу. Мы предполагаем, что использование языка ДРАКОН в сочетании с идеями Тавровского может оказаться плодотворным и успешным.

## ВЫВОДЫ

1. Следует различать одиночные алгоритмы и алгоритмические системы.
2. Одиночные алгоритмы полностью изолированы друг от друга; между ними нет связей.
3. *Медицинская алгоритмическая система* — это совокупность логически связанных медицинских алгоритмов, решающих общую задачу.
4. Понятие «алгоритмическая система» вводится для того, чтобы облегчить и сделать удобным проектирование больших и сложных алгоритмических комплексов.
5. Выигрыш состоит в том, что большой алгоритм разбивается на мелкие части и проектируется не целиком (что трудно), а по частям (что гораздо легче).
6. В языке ДРАКОН связь между алгоритмами осуществляется с помощью иконы Вставка.
7. При отсутствии Вставки говорить о наличии связи между алгоритмами бессмысленно и недопустимо.
8. Чтобы образовать связь между двумя алгоритмами, необходимо:
  - Предусмотреть в первом алгоритме икону Вставка.

- Надпись внутри Вставки должна совпадать с названием второго алгоритма.
  - Совпадение означает, что Вставка однозначно указывает на второй алгоритм.
  - В результате образуется связь между двумя алгоритмами, и они превращаются в алгоритмическую систему.
9. Алгоритмическая система обычно имеет иерархическую структуру. Она включает головной алгоритм и алгоритмы более низких уровней.
10. Головной алгоритм содержит, как правило, несколько икон Вставка. Каждая вставка указывает на одноименный алгоритм нижнего уровня.
11. Алгоритмы низких уровней также могут содержать Вставки, указывающие на другие алгоритмы.
12. Процесс проектирования алгоритма «сверху вниз» называется *декомпозицией* (разбиением на части) большого алгоритма.
13. В языке ДРАКОН для декомпозиции больших алгоритмов предусмотрены два инструмента:
- использование Вставок,
  - дробление алгоритмов на ветки.
14. Как правило, алгоритмическая система состоит из большого числа алгоритмов: от 10 до 100 и больше.
15. Примером большого алгоритма является алгоритмическая система скорой помощи.

## Глава 16

# КАКАЯ ПОЛЬЗА ОТ ЯЗЫКА ДРАКОН ДЛЯ ВРАЧЕЙ?

## КАК ИСПОЛЬЗОВАТЬ НА ПРАКТИКЕ ЯЗЫК ДРАКОН?

Можно указать ряд областей применения ДРАКОНа в медицине:

- альбомы-справочники медицинских алгоритмов;
- базы данных медицинских алгоритмов;
- электронная энциклопедия медицинских алгоритмов;
- медицинские стандарты;
- медицинские учебники и литература;
- медицинские информационные и экспертные системы.

## АЛЬБОМЫ-СПРАВОЧНИКИ

В качестве аналогов, образцов и предшественников можно назвать книги-справочники, содержащие графические изображения алгоритмов. Примерами являются:

- «Ваш семейный доктор. Домашний советчик» [75] [171].
- «Если ваш ребенок болен. Книга для родителей» [21].
- «Алгоритмы диагностики и лечения в хирургии» [77].
- «Неотложная педиатрия. Алгоритмы диагностики и лечения» [79] и др.

К сожалению, эти книги имеют серьезные недочеты. Они используют неунифицированные разноперые графические обозначения (созданные по принципу: кто во что горазд), не обеспечивают высокую точность алгоритмов, непригодны для полного контроля всех маршрутов. Их можно охарактеризовать как первое поколение алгоритмических справочников.

Язык ДРАКОН позволяет создать справочники второго поколения, содержащие стандартные и качественные комплекты профессиональных медицинских алгоритмов, свободные от указанных дефектов.

Серия альбомов-справочников на базе ДРАКОНа может быть разбита на отдельные книги по различным основаниям рубрикации, например, по специальностям: кардиология, пульмонология, гастроэнтерология и т. д.

## **ГЛОБАЛЬНАЯ ЭЛЕКТРОННАЯ БАЗА ДАННЫХ МЕДИЦИНСКИХ АЛГОРИТМОВ**

Наряду с бумажными медицинскими книгами все более широкое распространение получает электронная литература. Медицинские альбомы-справочники на языке ДРАКОН естественным образом будут представлены в цифровом варианте. Логично объединить электронные медицинские алгоритмы в единой базе данных.

По-видимому, сначала это будет база данных одного профессионального сообщества (или издательства), которое выступит в роли пионера новых идей. Затем возникнут общие (объединенные) базы данных двух или трех сообществ (издательств). И наконец, появится национальная база медицинских алгоритмов на русском языке. А там, глядишь, с развитием процесса глобализации со временем появятся глобальные базы данных.

Обязательным условием для развития этого процесса является стандартизация графического языка, т. е. использование единой графической нотации для записи медицинских алгоритмов. Язык ДРАКОН целенаправленно разрабатывался с учетом такой перспективы.

Для этого есть серьезная основа. Дело в том, что графический каркас дракон-алгоритмов совершенно не зависит от национального языка. Язык может быть любым: и русским, и английским, и китайским. Это значит, что графический узор дракон-алгоритма является надежным каркасом, «незыблемым оплотом», который сохраняется неизменным при переводе алгоритма с одного языка на другой.

Наличие «незыблемого оплата» существенно облегчает трансграничный обмен медицинскими алгоритмами. Радикальное упрощение проблемы перевода (например, с русского языка на английский или наоборот) связано с тем, что переводить надо лишь короткие надписи, находящиеся внутри икон, и больше ничего. Логика алгоритма, спрятанная в графическом скелете алгоритма, остается сохранной и не требует никаких изменений.

## **МЕЖДУНАРОДНЫЙ ПРОЕКТ «МЕДИЦИНСКИЙ АЛГОРИТМ»**

Выигрывает тот, кто умеет предвидеть будущее. Вдвойне выигрывает тот, кто способен это будущее создавать.

Попытаемся наметить контуры Проекта под названием «Медицинский алгоритм». Врачи всего мира, работающие на всех континентах, несомненно, желают иметь мгновенный доступ к международной базе медицинских алгоритмов. Сегодня такая база не существует, ее необходимо разработать и реализовать. В этом суть предлагаемого Проекта.

Включенные в базу алгоритмы должны удовлетворять очень строгим требованиям:

- соответствовать самым последним достижениям мировой медицинской науки,
- соответствовать требованию высокой точности,
- быть сертифицированными,
- быть представленными на родном языке врача.

Проект может опираться на уже достигнутые успехи и аналоги, такие как Интернет и Википедия.

Воодушевляющим примером успешной организации крупномасштабных работ является международный научно-исследовательский Проект «Геном человека» (The Human Genome Project).

## ОТ ВИКИПЕДИИ К АЛГОПЕДИИ

В качестве одного из возможных вариантов Проекта можно предложить создание новой, совершенно самостоятельной Глобальной электронной энциклопедии медицинских алгоритмов под условным название Алгопедия (Algopedia)<sup>14</sup>. Многие идеи Википедии могут оказаться полезными для Проекта. Наиболее простой и самый дешевый путь состоит в том, чтобы построить Алгопедию по образцу Википедии и, возможно, на основе ее движка MediaWiki [172].

Подобно бесплатной Википедии Алгопедия может:

- состоять из национальных разделов (русского, английского, немецкого и т. д.),
- создаваться добровольными участниками под контролем добровольцев-администраторов и сертификаторов, не получающих оплату за свой труд.

Как и Википедия, Алгопедия состоит из статей, обычно называемых *страницами*, или веб-страницами (web page). Веб-страница — документ или информационный ресурс Всемирной паутины, доступ к которому осуществляется с помощью веб-браузера [173]. Основным элементом каждой статьи Алгопедии является графический медицинский алгоритм, представленный на языке ДРАКОН.

Пользователи Алгопедии делятся на три категории:

- читатели,
- редакторы,
- сертификаторы.

Читатели — любые пользователи, они могут без регистрации читать все материалы, но не могут их редактировать.

Редакторы — дипломированные медики и, возможно, студенты медвузов. После регистрации на сайте они имеют право создавать статьи, предлагать новые медицинские алгоритмы и редактировать уже существующие, но не могут их сертифицировать.

Сертификаторы — наиболее авторитетные представители профессиональных сообществ, которым сообщество предоставило право сертификации алгоритмов.

Это примерный план, который, разумеется, нуждается в уточнении, исправлении и развитии. Например, права студентов медвузов можно ограничить. Можно позволить студентам создавать новые алгоритмы только в «песочнице» Алгопедии, а перевод статей из песочницы в основное пространство разрешить опытным участникам, скажем, их преподавателям.

---

<sup>14</sup> Чтобы подчеркнуть, что речь идет о медицинских алгоритмах, можно предложить расширенное название «АлгоМедипедия» (AlgoMedipedia).

Еще одно замечание. В русском разделе Алгопедии можно использовать такие же правила регистрации, как и на портале «Мир Врача», где для подтверждения статуса врача требуется «приложить скан-копию диплома или сертификата» [174].

Реализация проекта «Алгопедия» откроет перед медициной принципиально новые возможности, опирающиеся на информационные технологии XXI века. Преимущество состоит в том, что любой врач мира, имеющий доступ к интернету, может в любую минуту практически мгновенно получить информацию о любых интересующих его медицинских алгоритмах высокой точности.

## МЕДИЦИНСКИЕ СТАНДАРТЫ

Стандартизация — генеральная линия развития современной медицины. Этот процесс заметно ускорился после появления Международной Классификации Болезней. Документ является стандартом Всемирной организации здравоохранения, который непрерывно развивается и совершенствуется. Последняя версия документа известна как МКБ-10 (Классификация Десятого пересмотра) [175].

Что касается медицинских алгоритмов, то здесь наблюдается значительное отставание. Причина все та же — отсутствие общепринятой графической нотации для записи алгоритмов. Вследствие этого проблема алгоритмов выбивается из общего ритма медицинской стандартизации. Отсутствие удобной графической символики наносит огромный вред делу, тормозит прогресс и делает существующие решения неудовлетворительными.

Язык ДРАКОН предлагается как радикальное средство, способное обеспечить эффективную стандартизацию медицинских алгоритмов.

## ПРОБЛЕМА ОШИБОК В МЕДИЦИНСКИХ УЧЕБНИКАХ

Профессор Георгий Ратнер отмечает: «Всю профессиональную деятельность врача постоянно сопровождают ошибки диагностического, тактического и лечебного плана» [176].

По нашему мнению, значительная часть ошибок объясняется несовершенством медицинских учебников. Нас, разумеется, интересуют не любые ошибки, просочившиеся в учебники, а лишь те, что связаны с алгоритмами (назовем их алгоритмическими).

Существенную часть учебников составляют медицинские алгоритмы. Познавательная ценность учебников определяется их когнитивно-эргономическим качеством. Каково же это качество? Являются ли алгоритмы, представленные в медицинских учебниках сертифицированными? Кто осуществляет сертификацию?

Это риторические вопросы. Сегодня сертификация алгоритмов не проводится. Ответственность за качество учебников, учебных пособий, руководств и др. лежит на авторах и рецензентах. Иногда — на Ученом совете факультета, который одобрил учебник. Иногда ответственность оформляется более торжественно. Например, так:

Министерство образования и науки РФ

Рекомендовано ГБОУ ВПО «Первый Московский государственный университет имени И. М. Сеченова» в качестве учебника для студентов учреждений высшего профессионального образования, обучающихся по специальности 060101.65 «Лечебное дело» по дисциплине «Поликлиническая терапия».

Регистрационный номер рецензии 244 от 05 июня 2012 года ФГАУ «Федеральный институт развития образования» [200].

Достаточно ли этого? Вчера было достаточно.  
А сегодня уже нет, недостаточно. Чего же не хватает?

Не хватает умения заглянуть в будущее, чувства перспективы и требовательности. Чтобы кардинально уменьшить число врачебных ошибок и обеспечить безопасность пациентов, необходимо предъявить к медицинским учебникам, руководствам, клиническим рекомендациям и протоколам более строгие требования. **Необходима сертификация всех алгоритмов, представленных в учебниках и иных материалах и документах.**

## ПРОБЛЕМА СЕРТИФИКАЦИИ МЕДИЦИНСКИХ АЛГОРИТМОВ

Это трудная, очень сложная проблема. Ее предстоит решать в ближайшее время. Кто должен осуществлять сертификацию алгоритмов?

Готового ответа здесь нет. Учитывая международный опыт, думается, что производить сертификацию алгоритмов должны, по-видимому, профессиональные сообщества. В России они называются, как правило, научными обществами. Список (неполный) приведен ниже.

## ПЕРЕЧЕНЬ НАУЧНЫХ РОССИЙСКИХ МЕДИЦИНСКИХ ОБЩЕСТВ

- Ассоциация ревматологов России <http://www.rheumatolog.ru/>
- Ассоциация Флебологов России (АФР) <http://www.phlebo-union.ru/>
- Всероссийское Научное Общество Аритмологов (ВНОА) <http://www.vnoa.ru/>
- Всероссийское Научное Общество Кардиологов (ВНОК) <http://www.scardio.ru/>
- Всероссийское общество акушеров-гинекологов (Научный центр акушерства, гинекологии и перинатологии имени академика В.И. Кулакова) <http://www.ncagip.ru/for-experts/roag/>
- Национальное научно-практическое общество скорой медицинской помощи <http://cito03.ru/>
- Общество специалистов по сердечной недостаточности <http://www.ossn.ru/>
- Российская ассоциация аллергологов и клинических иммунологов (РААКИ) <http://www.raaci.ru/>
- Российская ассоциация специалистов перинатальной медицины (РАСМП) <http://www.raspm.ru/>
- Российская Гастроэнтерологическая Ассоциация <http://www.gastro.ru/>
- Российское диализное общество <http://www.nephro.ru/>
- Российское медицинское общество по артериальной гипертонии <http://www.gipertonik.ru/>
- Российское научное медицинское общество терапевтов <http://www.rnmot.ru/>
- Российское Общество ангиологов и сосудистых хирургов <http://www.angiolsurgery.org/>
- Российское общество психиатров <http://psychiatr.ru/>
- Российское общество фтизиатров <http://roftb.ru/>

- Российское Респираторное Общество (РРО) <http://www.pulmonology.ru/>
- Союз педиатров России <http://www.pediatr-russia.ru/>
- Федерация анестезиологов и реаниматологов (ФАР) <http://www.far.org.ru/>
- Эндокринологический научный центр (ЭНЦ)/ Российская Ассоциация Эндокринологов <http://www.endocrincentr.ru/>

## МЕДИЦИНСКИЕ АЛГОРИТМЫ. ПРИГОДНЫ ЛИ ОНИ ДЛЯ СЕРТИФИКАЦИИ?

Сертификация (подтверждение соответствия) есть «документальное удостоверение соответствия продукции или иных объектов... требованиям технических регламентов, положениям стандартов, сводов правил или условиям договоров» [177].

Приспособливая формулировку к нашему случаю, можно предложить уточнение. Сертификация медицинского алгоритма — документальное подтверждение соответствия алгоритма последним достижениям мировой медицинской науки, положениям международных стандартов о медицинских алгоритмах и сводам когнитивно-эргономических правил языка ДРАКОН. Поскольку нужные нам международные стандарты пока еще отсутствуют, приходится делать вывод, что существующие медицинские алгоритмы не соответствуют правилам ДРАКОНа и не пригодны для сертификации.

Вместе с тем вряд ли стоит ждать, пока будут расставлены все точки над i. Материалы, представленные в книге, являются достаточными для начала работы по совершенствованию медицинских алгоритмов и повышению их качества до уровня, пригодного для сертификации.

## ПРЕДПОЛАГАЕМАЯ МЕТОДИКА И ПРОЦЕДУРА

В качестве первого шага авторитетное медицинское сообщество должно дать заключение о правильности медицинского алгоритма. Или же указать на необходимость внесения уточнений и изменений.

Если через какое-то время, например, через год, появится новая информация о последних достижениях, то, по рекомендации сообщества, алгоритм должен быть исправлен.

После такой процедуры данный алгоритм (и десятки тысяч других медицинских алгоритмов) должны превратиться в постоянно обновляемый медицинский стандарт, отображающий самые последние достижения медицинской мысли.

Конечно, нынешняя реальность далека от нарисованной картины. Однако можно предположить, что скоро все изменится и мечта станет явью.

Тщательно разработанные эргономичные графические пошаговые алгоритмы высокой точности — будущее медицинской науки. Недалек день, когда одобренные авторитетными профессиональными сообществами медицинские алгоритмы (десятки тысяч алгоритмов) превратятся в постоянно обновляемые врачебные стандарты

## УПРОЩЕННЫЕ ВАРИАНТЫ ДЛЯ УЧЕБНЫХ ЦЕЛЕЙ

До сих пор мы вели речь о профессиональных медицинских алгоритмах. Однако в процессе обучения часто используется принцип «от простого — к сложному».

Сначала (например, на младших курсах) — исходя из дидактических соображений — полезно знакомить студентов с основной идеей лечебно-диагностического процесса, используя для иллюстрации упрощенные схемы алгоритмов. И постепенно, по мере накопления знаний, переходить ко все более подробным и сложным схемам.

Сколько же потребуется этапов усложнения? Сколько необходимо иметь вариантов одного и того же алгоритма? И какой вариант следует помещать в медицинский учебник? Это открытый вопрос, который нуждается в подробном обсуждении.

Профессор Александр Куликов на совещании у ректора РМАПО<sup>15</sup> академика Ларисы Мошетовой предложил двухступенчатую схему:

- упрощенный алгоритм для учебных целей,
- профессиональный (полный) алгоритм для работы.

Проректор РНИМУ<sup>16</sup> член-корреспондент Геннадий Порядин пошел еще дальше. По его мнению, на каждом курсе целесообразно давать студентам алгоритмы, отвечающие уровню их знаний и образующие цепочку постепенно нарастающей сложности.

## МЕДИЦИНСКИЕ ИНФОРМАЦИОННЫЕ И ЭКСПЕРТНЫЕ СИСТЕМЫ

Использование языка ДРАКОН в медицинских информационных и экспертных системах — отдельная и очень интересная тема. Но она выходит за рамки книги.

## ВЫВОДЫ

1. Предлагаются следующие области использования медицинских дракон-алгоритмов:
  - альбомы-справочники;
  - базы данных;
  - электронная энциклопедия Алгопедия;
  - медицинские стандарты;
  - медицинские учебники и литература.
2. Актуальной задачей является стандартизация формы представления медицинских алгоритмов.
3. Отсутствие удобной графической символики делает стандартизацию невозможной и наносит огромный вред делу.
4. Язык ДРАКОН предлагается как удобное средство, имеющее оптимальную графическую нотацию и способное обеспечить эффективную стандартизацию медицинских алгоритмов.

---

<sup>15</sup> Российская медицинская академия последипломного образования.

<sup>16</sup> Российский национальный исследовательский медицинский университет имени Н. И. Пирогова.

5. Язык ДРАКОН позволяет создать альбомы-справочники нового поколения, содержащие стандартные и качественные комплекты профессиональных медицинских алгоритмов.
6. Международная электронная энциклопедия медицинских алгоритмов «Алгопедия» откроет перед медициной принципиально новые возможности, опирающиеся на информационные технологии XXI века.
7. Преимущество Алгопедии состоит в том, что любой врач мира, имеющий доступ к интернету, может мгновенно получить информацию о любых интересующих его медицинских алгоритмах высокой точности.
8. Сертификация медицинских алгоритмов подразумевает:
  - контроль правильности алгоритмов и их соответствие последним достижениям мировой медицинской науки;
  - соответствие алгоритмов своду когнитивно-эргономических правил языка ДРАКОН.
9. Сертификацию медицинских алгоритмов целесообразно возложить на профессиональные медицинские сообщества.
10. Чтобы кардинально уменьшить число врачебных ошибок и обеспечить безопасность пациентов, желательно — в пределах возможного — проводить добровольную сертификацию всех алгоритмов, публикуемых в учебниках, руководствах, клинических рекомендациях, протоколах.
11. Материалы, представленные в книге, являются достаточными для начала работы по совершенствованию медицинских алгоритмов и повышению их качества до уровня, пригодного для сертификации.
12. Тщательно разработанные эргономичные графические пошаговые алгоритмы — ориентир для будущего.
13. Цель состоит в том, чтобы одобренные авторитетными профессиональными сообществами медицинские алгоритмы (десятки тысяч алгоритмов) превратились в постоянно обновляемые врачебные стандарты.
14. Наряду с профессиональными алгоритмами необходимо разработать упрощенные алгоритмы для учебных целей.

## Глава 17

# ДРАКОН-КОНСТРУКТОР. КАК СКЛАДЫВАТЬ АЛГОРИТМ ИЗ КУБИКОВ?

## ПОМОЩНИК ВРАЧА — АВТОМАТИЧЕСКИЙ РИСОВАЛЬЩИК

Разумеется, в случае крайней нужды дракон-схему можно нарисовать и вручную. Взять карандаш и набросать чертеж на листке бумаги. Однако это не лучший способ. Скорее всего, получится неаккуратно и некрасиво.

А если нужно что-то подправить? Придется зачеркивать линии и проводить их заново. Выйдет мазня. В конце концов, горе-художник убедится, что пошел по кривой дорожке и попал в тупик.

Разработка медицинского алгоритма требует труда и времени. Однако врач — человек занятой, и времени у него нет. Как быть?

Нужен шустрой слуга — автоматический рисовальщик, который поможет врачу быстро и без хлопот сочинить нужный алгоритм. Возьмите программу, которая называется ДРАКОН-конструктор. Это и есть автоматический рисовальщик, способный за считанные минуты изобразить красивую дракон-схему. Все мелкие дела робот-художник берет на себя, предоставляя врачу творческую часть работы.

## КОНСТРУКТОР АЛГОРИТМОВ

Уточним терминологию. В этой главе кратко описывается компьютерная программа «Конструктор алгоритмов». Она представляет собой рабочий инструмент, помогающий человеку придумывать и конструировать алгоритмы.

Поскольку нас интересуют медицинские алгоритмы, вместо выражения «конструктор алгоритмов» предлагается термин *ДРАКОН-конструктор*. Последний предназначен для разработки и оформления медицинских алгоритмов.

## ПРАВИЛА ДРАКОНА

Язык ДРАКОН содержит большое число правил. К счастью, их не надо учить и запоминать. Потому что правила знает назубок, хранит в своей обширной памяти и скрупулезно выполняет компьютерная программа ДРАКОН-конструктор.

Вот пример. В дракон-схемах запрещено пересечение линий. Запрет выполняется автоматически. ДРАКОН-конструктор гарантирует отсутствие пересечений. Рисунки в этой книге подтверждают правило. Пересечения линий нигде и никогда не используются.

## ГДЕ СКАЧАТЬ ДРАКОН-КОНСТРУКТОР?

К услугам врачей имеется не одна, а несколько бесплатных программ, каждая из которых умеет рисовать медицинские дракон-алгоритмы.

Начать знакомство лучше всего с программы «DRAKON Editor Web» (автор Степан Митькин).

<http://drakon-editor.com> Чтобы попробовать ее в деле, нажмите на кнопку «Try me».

Это онлайн программа. Ее не надо скачивать, она работает в браузере.

Вторая программа называется «ИС Дракон», разработчик Геннадий Тышов. Скачать можно здесь:

[http://drakon.su/programma\\_is\\_drakon](http://drakon.su/programma_is_drakon)

Третья программа «Фабула», автор Эдуард Ильченко:

<http://forum.oberoncore.ru/viewtopic.php?f=79&t=5475>

Четвертая и последняя программа «DRAKON Editor». Как и первая, она создана Степаном Митькиным <https://sourceforge.net/projects/drakon-editor/files/>

Программы непрерывно развиваются и дополняются. Не исключено, что к моменту выхода книги на сайте и форуме ДРАКОНа появится информация о создании новых, более совершенных вариантов ДРАКОН-конструктора.

## ГДЕ ПОЛУЧИТЬ ИНТЕРНЕТ-КОНСУЛЬТАЦИИ?

Если что-то непонятно, можно получить помощь на Официальном форуме языка ДРАКОН. После обычной регистрации вы можете задавать любые вопросы непосредственно авторам программ «DRAKON Editor Web», «ИС Дракон», «Фабула» и участвовать в обсуждении:

<http://forum.oberoncore.ru/viewforum.php?f=77>

Официальный сайт «Визуальный язык ДРАКОН» находится здесь:

<http://drakon.su/>

## ВИДЕО И ПРЕЗЕНТАЦИИ

Возможно, кто-нибудь захочет быстро получить общее представление о языке ДРАКОН и его бойких слугах-рисовальщиках. Лучше всего это сделать с помощью видео и слайд-шоу:

[http://drakon.su/video\\_i\\_presentation/start](http://drakon.su/video_i_presentation/start)

Загляните также в Википедию, где статья «ДРАКОН» представлена:

- на русском языке <http://ru.wikipedia.org/?oldid=74470829>
- на английском языке <https://en.wikipedia.org/w/index.php?title=DRAKON&oldid=700336070>
- на французском языке <http://fr.wikipedia.org/w/index.php?title=DRAKON&oldid=123191815>

Заодно можно посмотреть вики-статью «Медицинский алгоритм»:

<http://ru.wikipedia.org/?oldid=73589154>

## ГРАФИЧЕСКОЕ МЕНЮ

В состав ДРАКОН-конструктора входит графическое меню. Также его называют **панель инструментов** (toolbar) (рис. 113). Чтобы нарисовать дракон-схему, пользователь сначала вызывает меню на экран компьютера. А затем с его помощью рисует или, как говорят, конструирует схему.

Элементы меню называются **кубики**. Меню содержит 18 элементов.

Надписи на рис. 113 (Заготовка-силуэт, Заготовка-примитив, Пауза и т. д.) оформляются как всплывающие подсказки (tooltips). Они появляются на экране лишь тогда, когда курсор мыши подводится к выбранному вами кубику.

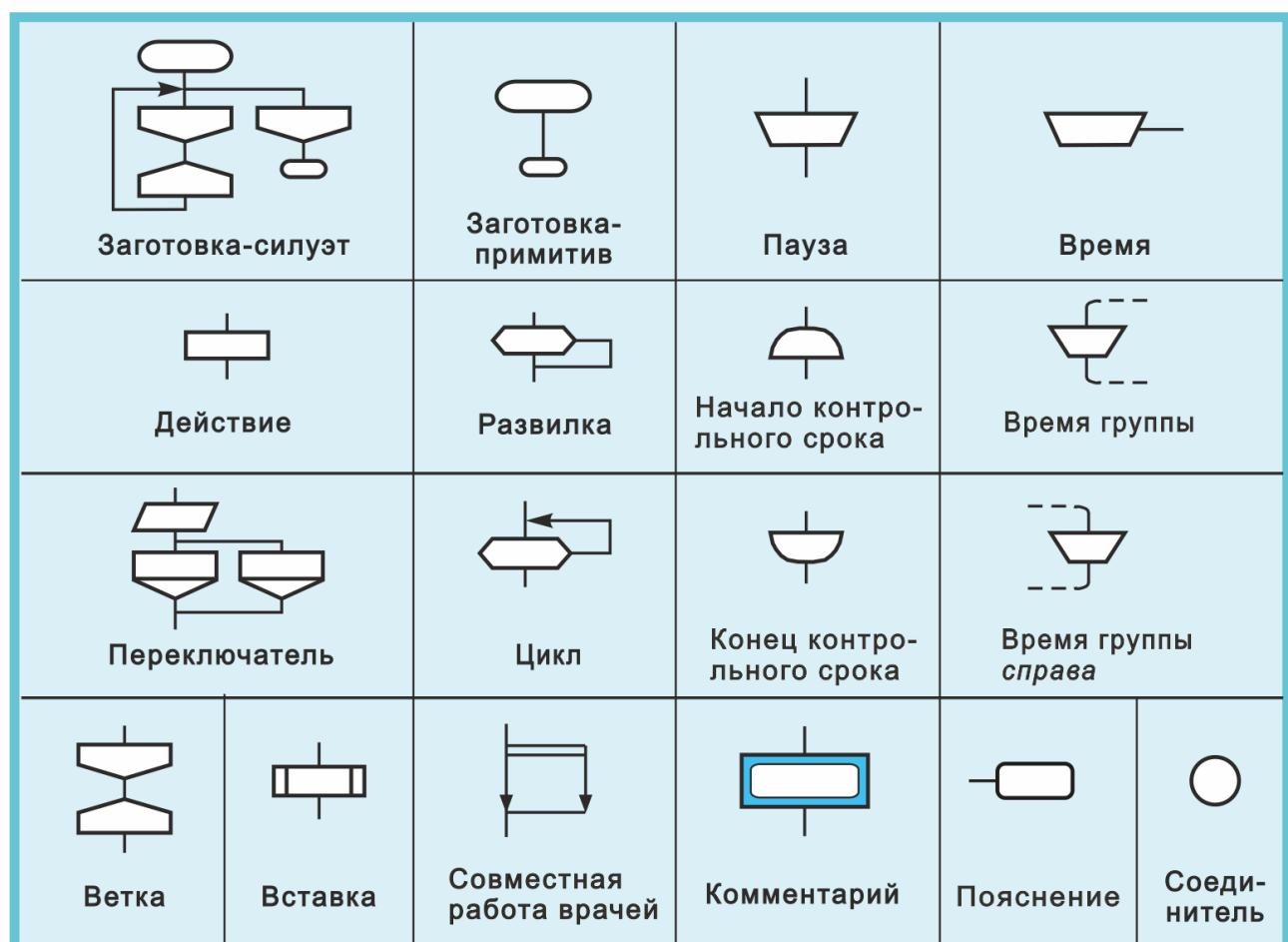


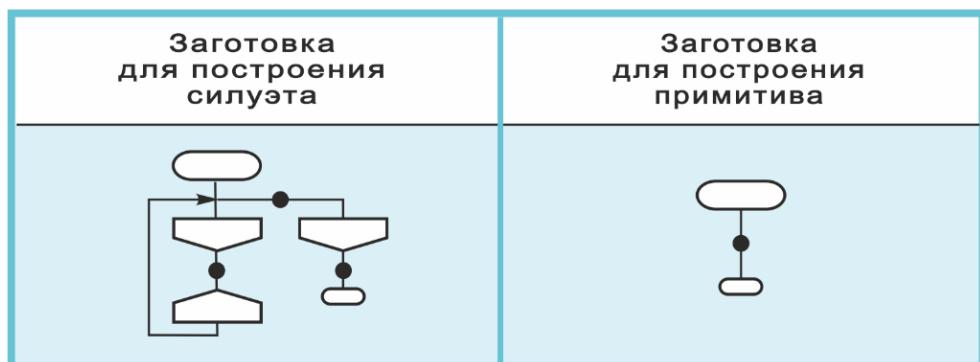
Рис. 113. Графическое меню ДРАКОН-конструктора

## ЗАГОТОВКА-СИЛУЭТ И ЗАГОТОВКА-ПРИМИТИВ

Чтобы вырастить огромное дерево, нужно бросить в землю маленько семечко. Любая сколько угодно сложная дракон-схема тоже вырастает из семечка, которое называется заготовкой.

Заготовки бывают двух сортов. Одна (заготовка-силуэт) используется для построения силуэта. Из другой получается примитив (рис. 114).

Заготовка содержит валентные точки (черные кружки). В них можно вставлять кубики из меню на рис. 113.



**Рис. 114.** Преобразуя заготовки с помощью простых визуальных операций, можно построить любую дракон-схему

## СБОРКА ИЗ КУБИКОВ

Графическое меню упрощает работу пользователя. Оно дает возможность рисовать схему методом «сборки из кубиков». Для этого служит операция «Вставь кубик», позволяющая доставать кубики из меню и укладывать их в проектируемую схему.

Кубики вставляются не куда попало, а только в разрешенные места, которые называются *валентными точками*.

## ЗАДАЧА: ПОСТРОИТЬ ПРИМИТИВ ПО ЗАДАННОМУ ОБРАЗЦУ

Как работает ДРАКОН-конструктор? Каким образом он строит дракон-алгоритм? Как это можно пощупать и попробовать на зуб?

Чтобы узнать вкус яблока, надо его съесть. Лучше всего взять одну из компьютерных программ (например, DRAKON Editor Web) и научиться с ней работать. В дополнение к практическому опыту дадим пояснения — расскажем о механизме построения алгоритма с помощью ДРАКОН-конструктора.

Сделаем это, разбирая конкретный пример.

Предположим, имеется задача: необходимо построить Примитив, изображенный на рис. 115. Изобразить в точности, без каких-либо отступлений от чертежа.

Как приступить к делу? С чего начать? Куда запрягать лошадь?

Мы уже знаем: чтобы построить дракон-схему «Примитив», нужно выращивать ее из семечка. Семечком является заготовка-примитив. Стало быть, начать нужно с заготовки. Имея заготовку, будем постепенно, шаг за шагом, превращать ее в образец на рисунке 115.

## НЕ ЦАРСКОЕ ЭТО ДЕЛО

Здесь нас ждет приятный сюрприз. Забудьте о линиях! Не царское это дело — рисовать линии! Все линии аккуратно, без ошибок и гораздо лучше вас нарисует ваш покорный слуга ДРАКОН-конструктор. Он нарисует их автоматически и моментально. На вашу долю остается самая легкая и приятная работа — вставлять кубики. И больше ничего.

Как только вы добавите в схему очередной кубик, умница-конструктор сразу «присобачит» к нему все необходимые линии.

## КАК ВСТАВИТЬ КУБИК

Команда «Вставь кубик» выполняется за два щелчка мыши:

- щелкаем нужный кубик в меню,
- затем обращаемся к дракон-схеме и указываем точку, куда нужно его вставить.

При этом происходит разрыв соединительной линии в выбранной пользователем валентной точке. В место разрыва вставляется кубик.

Как же это делается? Как кубик укладывается в схему? Как выбирается точка для внедрения кубика?

Подробные ответы даны на рисунке 116, где приведены восемь поучительных примеров. Глядя в среднюю графу, можно проследить, как в схему проникают следующие кубики:

- Действие.
- Вставка.
- Развилка.
- Переключатель.

Где находятся валентные точки? На рис. 116 они изображены в левой графике как маленькие черные кружки. Мы видим, что кружки могут находиться в разных местах:

- в заготовке-примитив;
- в развилке (и в левом, и в правом плече);
- между двумя иконами Действие;
- в цикле;
- в переключателе.

В реальных дракон-схемах валентные точки не изображаются, а подразумеваются.

Изучение приемов, представленных на рис. 116, позволяет приступить к нашей основной задаче и ответить на вопрос: *Как нарисовать дракон-схему «Примитив» на рис. 115.*

Чтобы подготовить рабочее место, вызываем на экран графическое меню (рис. 113). Затем выбираем кубик «Заготовка-примитив» и помещаем его в рабочее поле экрана.

## ПРИМЕР ПОСТРОЕНИЯ ДРАКОН-СХЕМЫ «ПРИМИТИВ»

Решение задачи показано в виде семи последовательных шагов на двух рисунках: 117 (шаги 1–4) и 118 (шаги 5–7). Результатом является точная копия образца на рис. 115. Вы найдете этот результат на рис. 118 справа внизу (см. шаг 7). Сравнив результат с рисунком 115, можно убедиться, что рисунки совпадают и задача решена правильно.

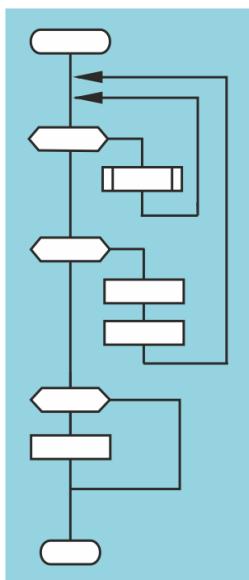
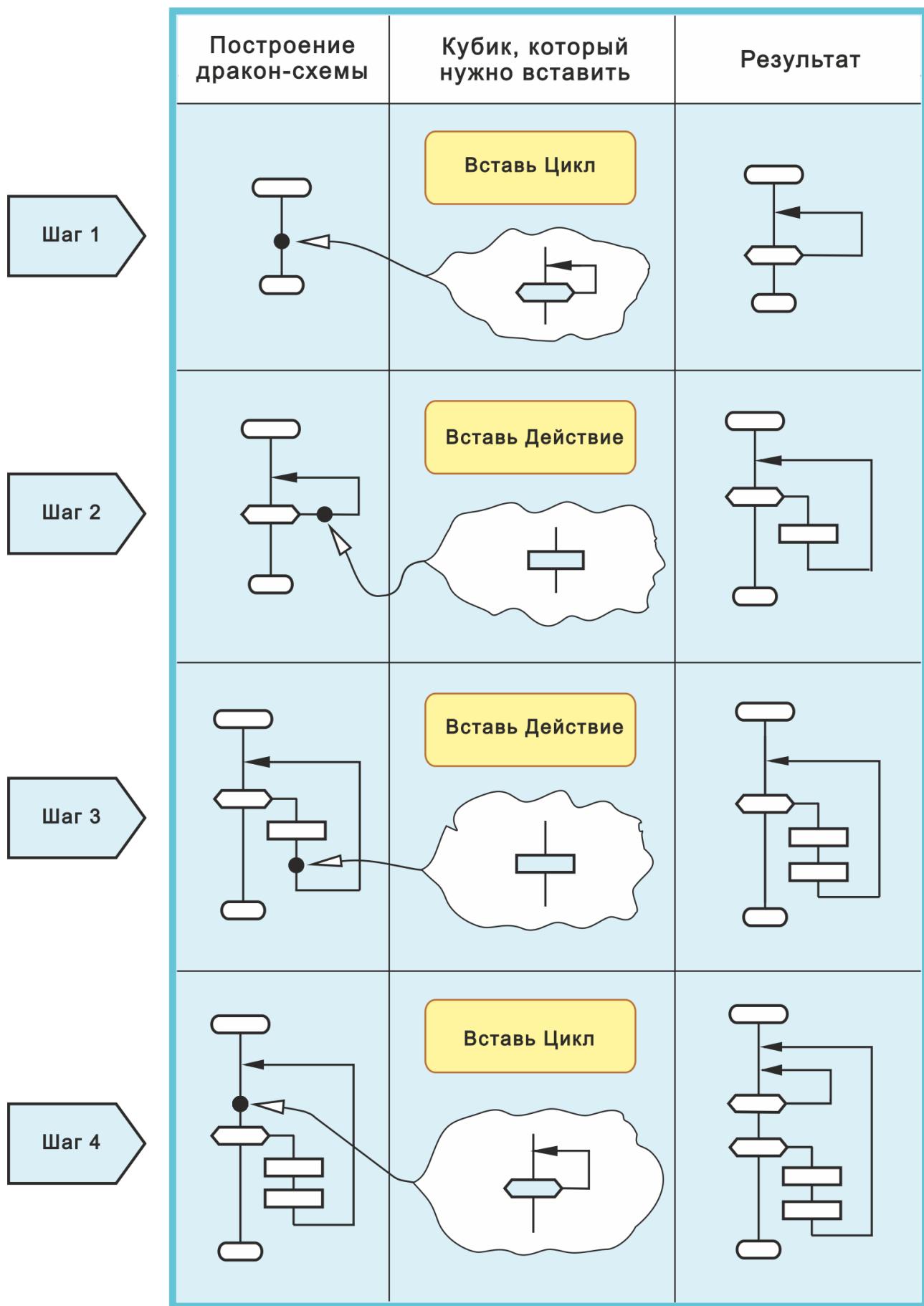


Рис. 115. Задача: нужно построить вот такой Примитив

	Исходная схема	Кубик, который нужно вставить	Результат
Пример 1			
Пример 2			
Пример 3			
Пример 4			
Пример 5			
Пример 6			
Пример 7			
Пример 8			

Рис. 116. Примеры выполнения операции «Вставь кубик»



**Рис. 117.** Конструирование дракон-схемы «Примитив» с помощью дракон-конструктора

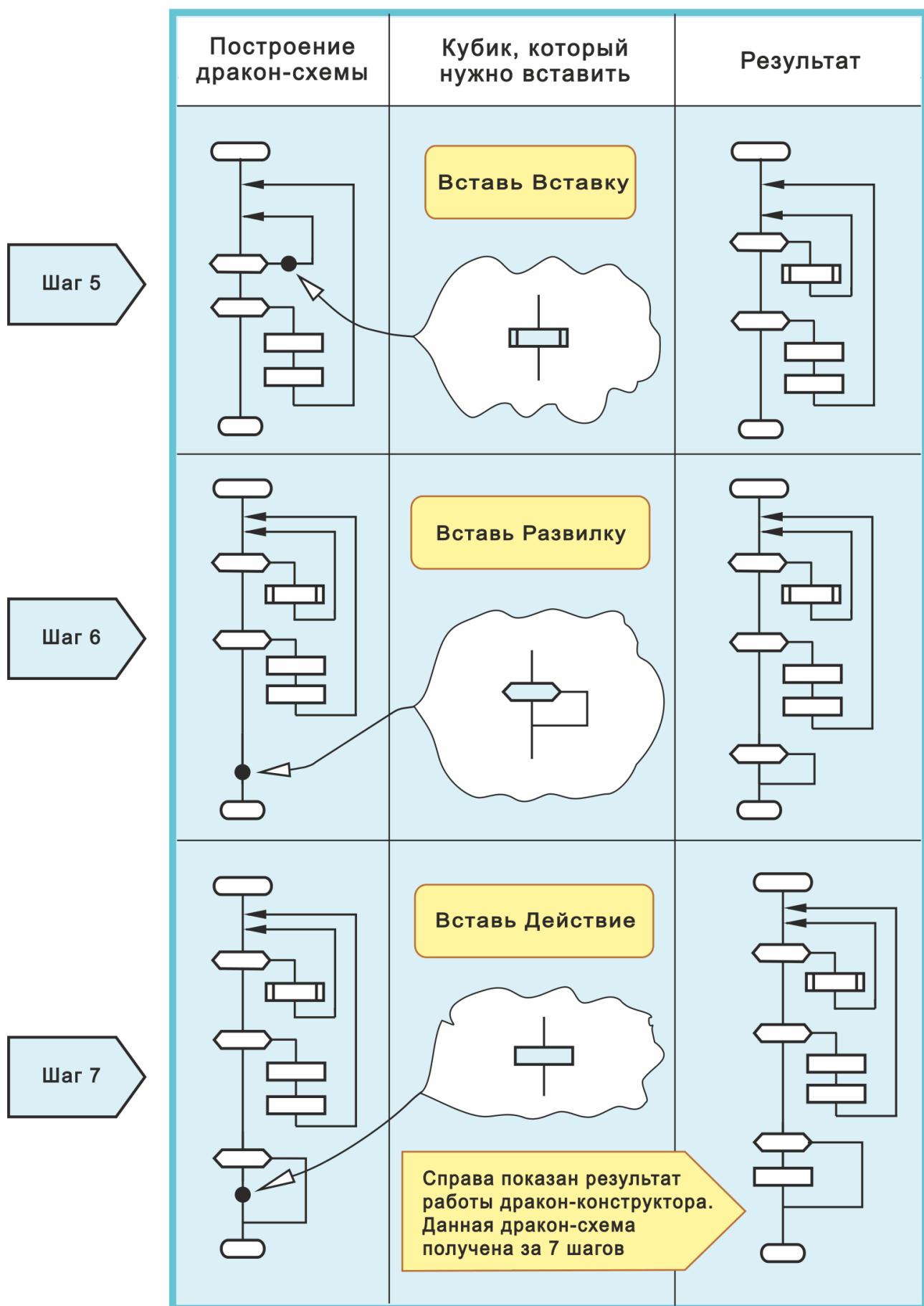


Рис. 118. Конструирование дракон-схемы «Примитив» (окончание). Справа внизу показана схема, созданная по образцу на рис.115.

Итак, приступим.

На первом шаге вызываем из меню кубик Цикл (см. среднюю графу на рисунке 117, *шаг 1*).

Куда его поместить? Подводим курсор к нужной точке в заготовке-примитив. К той самой точке, в которой следует разорвать соединительную линию, чтобы в образовавшийся разрыв вставить выбранный кубик. Результат операции виден на рисунке 117, *шаг 1, справа*.

Два следующих шага выполняем аналогично. В схему последовательно вставляем две иконы Действие (рис. 117, *шаги 2 и 3*).

Далее следует ввод кубика Цикл (рис. 117, *шаг 4*).

Чуть-чуть передохнем и подведем предварительный итог. О чём говорит нам рисунок 117? Что же мы сделали за первые четыре шага? В чём состоит сухой остаток?

Глядя на среднюю графу, убеждаемся, что мы выполнили четыре команды:

- Вставь Цикл.
- Вставь Действие.
- Вставь Действие.
- Вставь Цикл.

Кроме того (если глянуть на черные кружки в левой графе), можно заметить, что для каждой команды мы аккуратно указали точку, куда надо вставить очередной кубик.

Но самое интересное, конечно, находится в правой графе. Изучая ее сверху вниз, мы видим, как постепенно и неуклонно на наших глазах растет драгоценное сокровище — дракон-схема «Примитив», которая и является нашей желанной целью.

Итог наших трепетных усилий на рисунке 117 находится в правом нижнем углу. Он стоит того, чтобы посмотреть на него внимательно.

Едем дальше и переводим взгляд на рисунок 118. По очереди вставляем в схему кубики Вставка, Развилка и Действие. Здесь нам уже все знакомо. На всякий случай, подытожим. Глядя на среднюю графу, замечаем, что мы выполнили три команды:

- Вставь Вставку.
- Вставь Развилку.
- Вставь Действие (рис. 118, *шаги 5–7*).

Хороший следопыт, посмотрев на левую графу и проследив черные кружки, легко догадается, в какие валентные точки попали наши меткие кубики.

Что ж, на этом можно закончить. Результат последней операции показан на рис. 118 (*шаг 7, справа*). Этот результат достойно венчает наши труды, потому что он в точности совпадает с заданием на рис. 115.

После того, как графический узор (слепыш) дракон-схемы построен, производится заполнение его текстом.

## ЧТО ТАКОЕ ЛИАНА

Обезьяна, сидевшая на дереве, поймала свисавшую сверху лиану. Однако нижняя часть лианы приросла к стволу и не поддавалась. Обезьяна перегрызла ее зубами, уцепилась за конец и мигом перелетела на соседнее дерево, где прочно привязала лиану к ветке.

Нечто подобное умеет делать и дракон-конструктор.

Лиана — часть дракон-схемы, которая:

- имеет начало и конец (начало лианы и конец лианы);
- начало лианы находится вверху, конец — внизу;
- от начала к концу проходит хотя бы один маршрут;

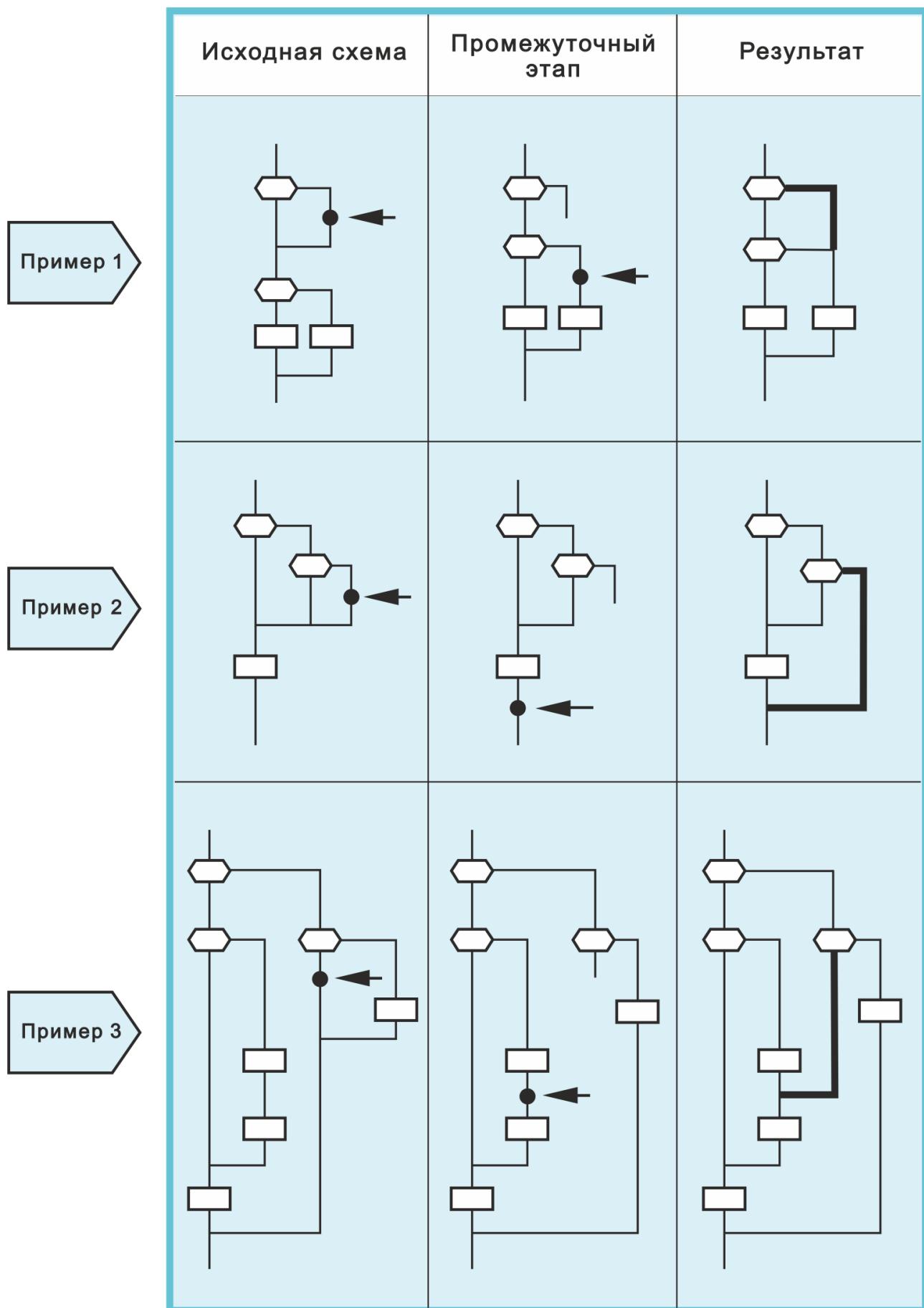


Рис. 119. Примеры выполнения операции «Пересадка лианы»

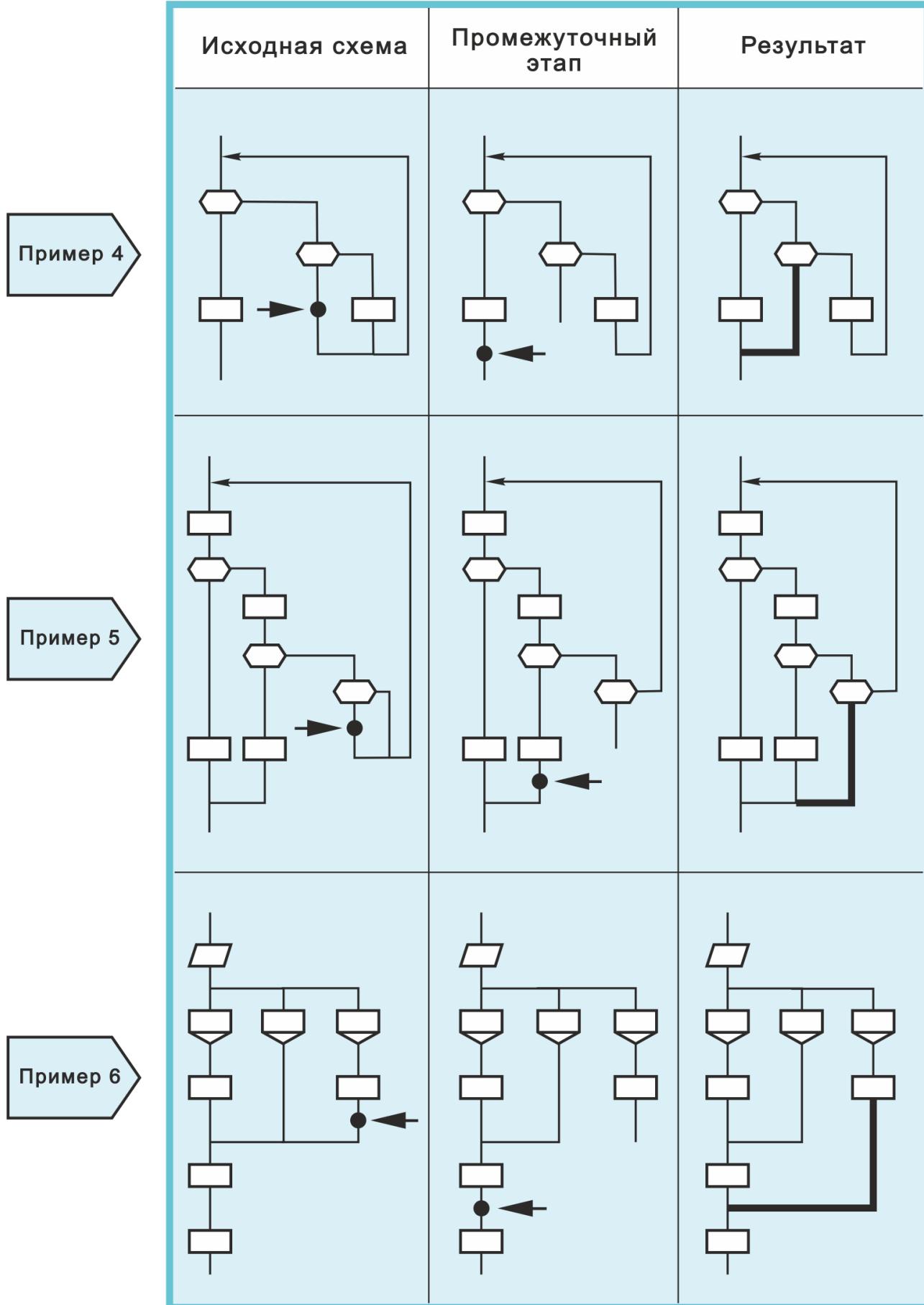


Рис. 120. Примеры выполнения операции «Пересадка лианы» (окончание)

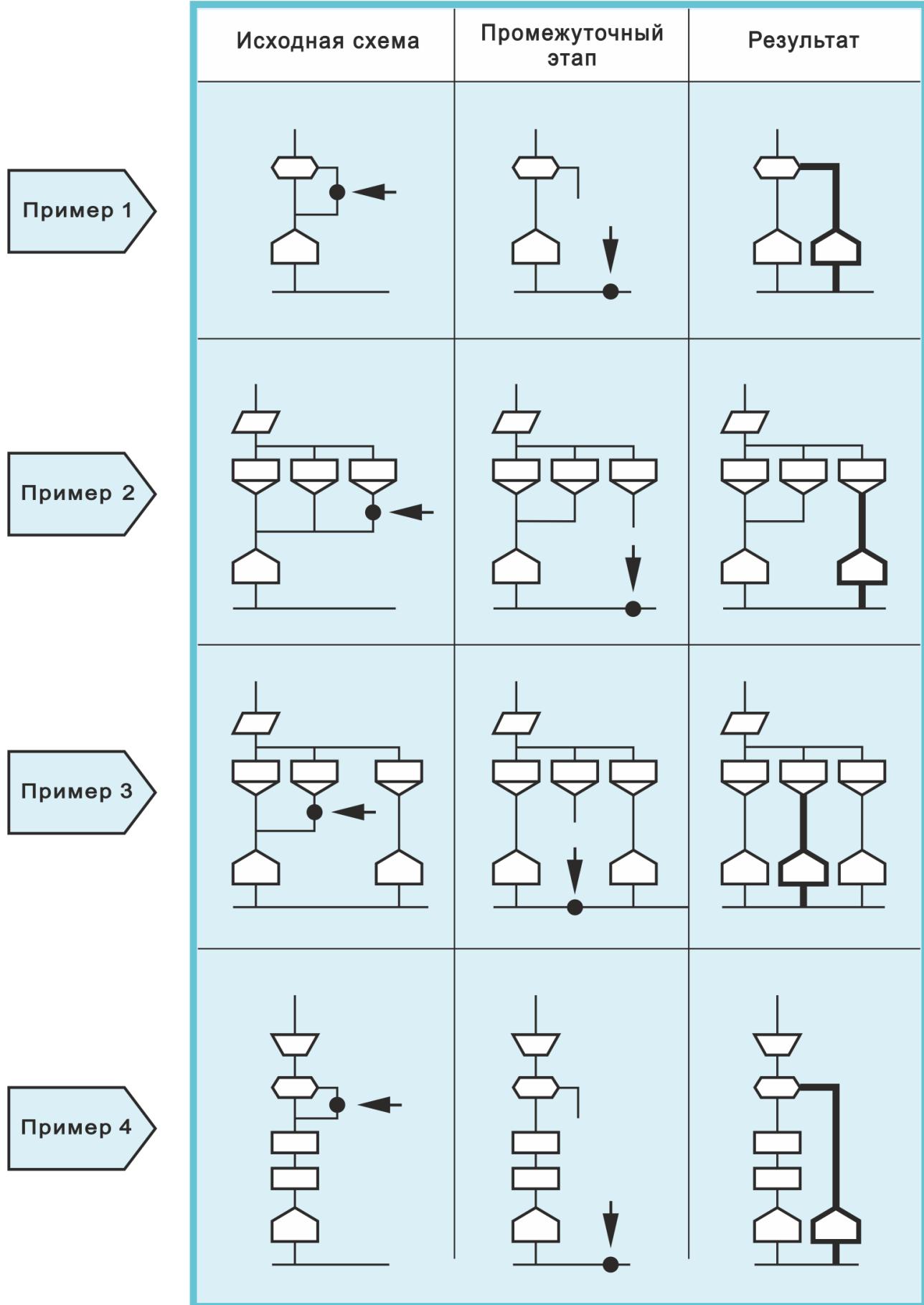


Рис. 121. Примеры выполнения операции «Заземление лианы»

- началом лианы служит выход иконы Вопрос или Вариант (если этот выход не является петлей цикла);
- концом лианы служит точка слияния.

Обычно лиана — это просто соединительная линия. Но иногда она может содержать одну или несколько икон.

## ОПЕРАЦИЯ «ПЕРЕСАДКА ЛИАНЫ»

Иногда бывает нужно оторвать конец лианы от своего места и присоединить в другую точку схемы. Такая операция называется *пересадка лианы*. Примеры показаны на рис. 119 и 120.

Операция выполняется в два этапа. Сначала курсор подводится к лиане, конец которой надо освободить (рис. 119, *левая графа*).

Куда его присоединить? Выбираем желаемую точку и отмечаем ее курсором (рис. 119, *средняя графа*). Результат операции «Пересадка лианы» показан на том же рисунке в правой графе.

Что показано в Примере 1? Отрываем правое плечо от верхней развилки и присоединяем его (без пересечения линий!) к нижней развилке.

Чему учит нас Пример 2? Отрываем правое плечо от правой развилки и присоединяем линию (без пересечений!) в валентную точку ниже иконы Действие.

Пример 3 тоже очень интересен. Он показывает, что левое плечо развилки также можно оторвать и (без пересечений!) присоединить к валентной точке.

На рис. 120 представлены дополнительные примеры пересадки лианы. Они демонстрируют разнообразие используемых приемов.

Многие дракон-схемы, представленные в книге, построены с помощью пересадки лианы.

## ОПЕРАЦИЯ «ЗАЗЕМЛЕНИЕ ЛИАНЫ»

Пересадка лианы применима и к примитиву, и к силуэту. В отличие от нее заземление лианы относится только к силуэту. Оно служит для построения веток, имеющих несколько выходов (многоадресных веток).

Чтобы заземлить лиану, необходимо:

- организовать в ветке разветвление (с помощью Развилки или Переключателя);
- оторвать присоединенную к ним лиану от прежнего места;
- присоединить ее через икону Адрес к нижней горизонтальной линии силуэта, то есть «заземлить» ее.

При заземлении лианы различают два этапа.

Первый этап (отрыв нижнего конца лианы от своего места) осуществляется так же, как при пересадке лианы (рис. 121, *левая графа*).

На втором этапе пользователь подводит курсор к нижней линии силуэта, указывая точку, куда лиана может дотянуться, не пересекая других линий (рис. 121, *средняя графа*).

Это действие порождает автоматическое появление в нужном месте иконы Адрес. Лиана автоматически присоединяется к иконе Адрес. И через нее — к нижней линии силуэта (рис. 121, *правая графа*).

Заземление лианы используется при построении силуэтов с многоадресными ветками, как показано на рисунках 92 и 102.

## ЗАДАЧА: ПОСТРОИТЬ СИЛУЭТ ПО ЗАДАННОМУ ОБРАЗЦУ

Как работает ДРАКОН-конструктор? Каким образом он строит Силуэт? Как это происходит?

Силуэт — основной инструмент языка ДРАКОН, обладающий огромным арсеналом выразительных средств. Мы знаем, что силуэт способен изображать очень сложные алгоритмы, такие как:

- Первая помощь при химическом ожоге глаз (рис. 96).
- Реанимация беременной женщины (рис. 102).

Однако мы ничего не знаем о тех таинственных механизмах, которые использует ДРАКОН-конструктор при вычерчивании Силуэта. Пора прояснить этот вопрос и рассказать о секретах ДРАКОН-конструктора.

Сделаем это, рассматривая конкретный пример.

Предположим, имеется задача: необходимо построить Силуэт, изображенный на рис. 122. Иными словами, на рис. 122 задан образец, который мы должны воспроизвести с помощью конструктора.

Как приступить к делу? Чтобы построить дракон-схему «Силуэт», нужно выращивать ее из семечка, т. е. из заготовки-силуэт. Стало быть, начать нужно с заготовки.

Исходя из этого, мы привели заготовку-силуэт в состояние боевой готовности и поместили ее туда, куда нужно. Она ждет своего часа на рис. 123 в левом верхнем углу. Имея заготовку, будем постепенно, шаг за шагом, превращать ее в образец на рисунке 122.

## ПРИМЕР ПОСТРОЕНИЯ ДРАКОН-СХЕМЫ «СИЛУЭТ»

Решение задачи представлено в виде десяти последовательных шагов на трех рисунках: 123 (шаги 1–4), 124 (шаги 5–7) и 125 (шаги 8–10).

Результатом является точная копия образца на рис. 122. Этот результат находится на рис. 125 в правом нижнем углу (см. шаг 10). Сравнив результат с рисунком 122, можно удостовериться, что они полностью совпадают. И, значит, задача решена правильно.

Что ж, «начнем, пожалуй», как говорил Ленский перед дуэлью.

На первом шаге вызываем из меню кубик Ветка (см. среднюю графу на рисунке 123, шаг 1).

Куда его поместить? Подводим курсор к нужной точке в заготовке-примитив. К той самой точке, в которой следует разорвать соединительную линию, чтобы в образовавшийся разрыв вставить выбранный кубик. Тем самым мы изменяем заготовку, добавляя в нее еще одну ветку. Результат операции виден на рисунке 123, шаг 1, справа.

Три следующих шага выполняем аналогично. В схему последовательно вставляем три иконы Действие (рис. 123, шаги 2–4).

Подведем предварительный итог. О чём говорит рисунок 123? Что сделано за первые четыре шага? Каков сухой остаток?

Глядя на среднюю графу, убеждаемся, что мы выполнили четыре команды:

- Вставь Ветку.

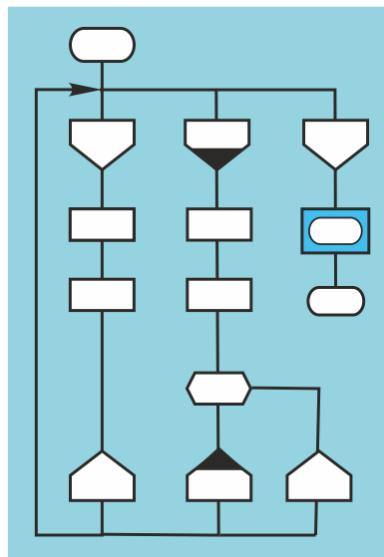


Рис. 122. Задача: нужно построить такой силуэт

- Вставь Действие.
- Вставь Действие.
- Вставь Действие.

Черные кружки в левой графе показывают, что для каждой команды мы заботливо указали точку, куда надо вставить очередной кубик.

Обратимся теперь к самому интересному — к правой графе. Просматривая ее сверху вниз, мы видим, как постепенно, шаг за шагом, растет и обрастает новыми деталями дракон-схема «Силуэт», которая и является нашей целью.

Итог наших стараний на рисунке 123 находится в правом нижнем углу. На данном этапе он похож на недостроенное здание и состоит всего из 10 икон.

Идем дальше и переводим взгляд на рисунок 124. По очереди вставляем в схему кубики Действие, Развилка и начинаем заземлять лиану. Начинать-то начинаем, но пока не заканчиваем. Мы успели лишь оторвать нижний конец лианы. Глядя в правый нижний угол, видим, что оторванный конец сиротливо висит и ждет своей участи.

Подытожим. Глядя на среднюю графу, замечаем, что мы выполнили две команды:

- Вставь Действие.
- Вставь Развилку.

После этого мы наполовину выполнили заземление лианы (рис. 118, шаги 5–7).

Переходим к рисунку 125. Здесь мы вместе со слугой-конструктором совершили три подвига:

*Подвиг 1.* Доведено до конца заземление лианы. Пользователь щелкнул мышью по нижней линии силуэта, а умница-конструктор мигом все понял и автоматически:

- вставил икону Адрес;
- облепил икону Адрес всеми необходимыми линиями.

*Подвиг 2.* Проставлены маркеры веточного цикла (черные треугольники) в иконы «Имя ветки» и «Адрес». Этот подвиг умница-конструктор выполняет полностью самостоятельно, без вмешательства человека, так что пользователь даже мизинцем не пошевелил. Оговоримся: конструктор автоматически выполняет эту операцию, анализируя текст в иконах «Имя ветки» и «Адрес». При отсутствии текста он ничего не делает. Так что в нашем «бессловесном» примере на рис. 122 простановка маркеров невозможна; она выполняется только после расстановки надписей в иконах.

*Подвиг 3* более чем скромен. Это всего лишь выполнение заурядной команды «Вставь Комментарий».

На этом летопись подвигов ДРАКОН-конструктора можно закончить. Результат последней операции показан на рис. 125 (*шаг 10, справа*). Этот результат завершает дело, потому что он в точности совпадает с заданием на рис. 122.

После того, как графический узор (слепыш) дракон-схемы построен, производится заполнение его текстом. Только после этого ДРАКОН-конструктор совершает свой «подвиг» и производит расстановку маркеров веточного цикла.

## ФОРМИРОВАНИЕ НАДПИСЕЙ «ДА» И «НЕТ»

Возле каждой иконы Вопрос обязательно должны стоять надписи «Да» и «Нет». Эти надписи появляются на дракон-схеме всякий раз, когда из меню вызывается элемент, содержащий икону Вопрос (рис. 113).

Первоначально ДРАКОН-конструктор пишет «Да» у нижнего выхода иконы Вопрос и «Нет» — у правого. Пользователь может менять их местами. Для этого в конструкторе предусмотрена операция «Да/Нет». При выполнении этой операции,

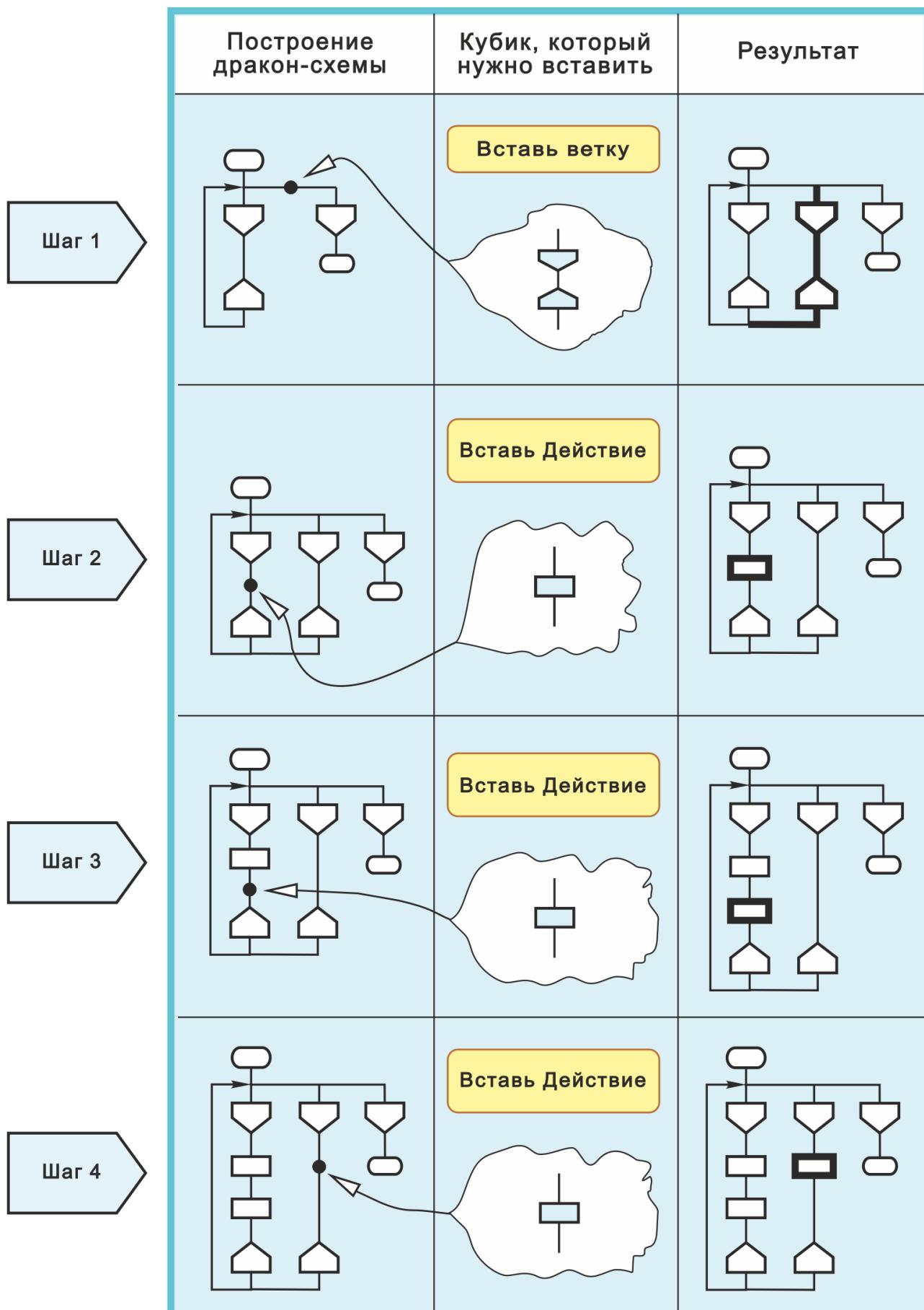
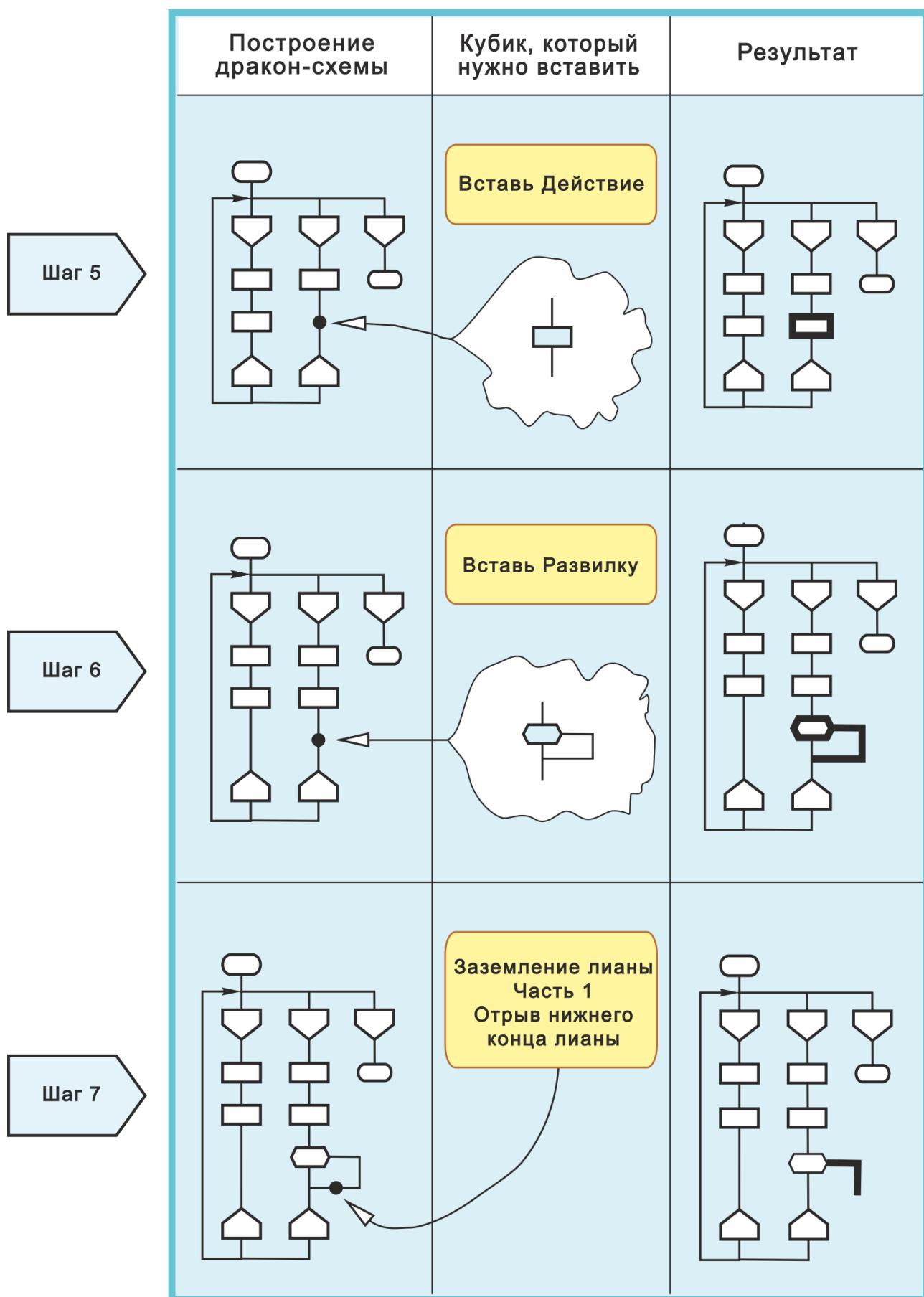
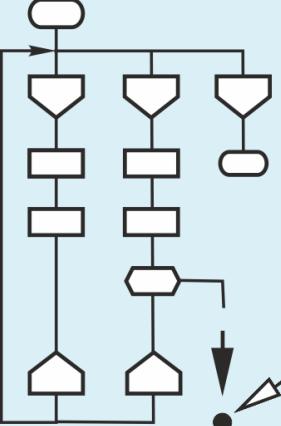
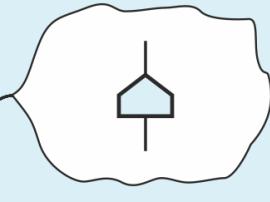
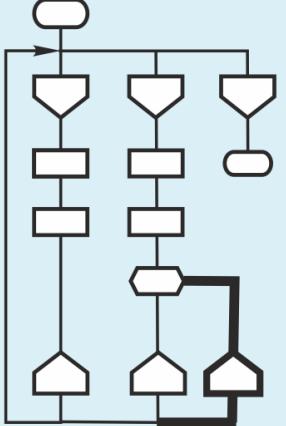
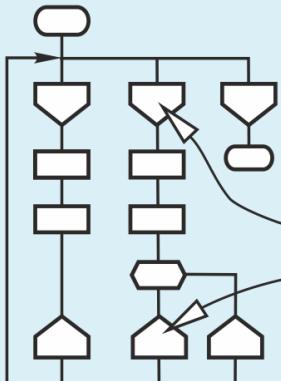
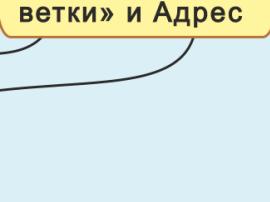
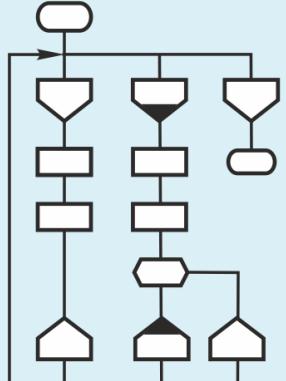
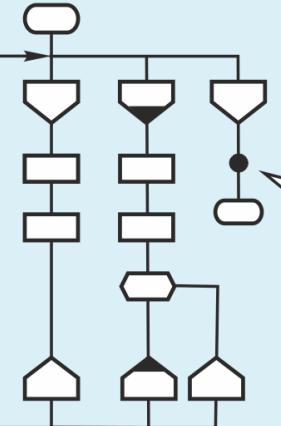
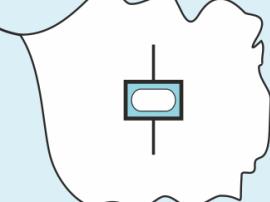
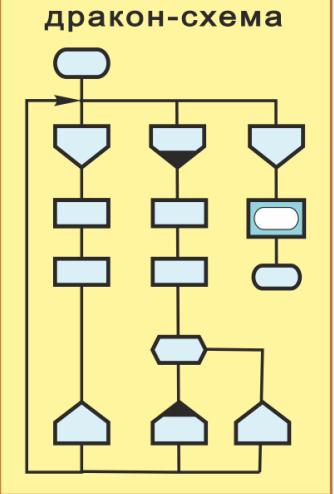


Рис. 123. Конструирование дракон-схемы «Силуэт» с помощью дракон-конструктора



**Рис. 124.** Конструирование дракон-схемы «Силуэт» с помощью дракон-конструктора (продолжение)

Построение дракон-схемы	Кубик, который нужно вставить	Результат
Шаг 8	 <p><b>Заземление лианы Часть 2 Автоматический ввод иконы Адрес</b></p> 	
Шаг 9	 <p><b>Ввод маркера цикла (черный треугольник) в две иконы: «Имя ветки» и Адрес</b></p> 	
Шаг 10	 <p><b>Вставь Комментарий</b></p> 	<p><b>Окончательная дракон-схема</b></p> 

**Рис. 125.** Конструирование дракон-схемы «Силуэт» (окончание). Справа внизу показана схема, созданная по образцу на рис.122.

слова «Да» и «Нет» у выходов иконы Вопрос меняются местами. (При этом все остальные элементы схемы остаются на своих местах).

Многократное нажатие на кнопку «Да/Нет» приводит к тому, что «Да» и «Нет» поочередно меняют свое положение.

## ЧЕМ ОТЛИЧАЕТСЯ ОПЕРАЦИЯ «ДА/НЕТ» ОТ РОКИРОВКИ?

При рокировке алгоритм не меняется. Потому что — вместе со сменой «Да» и «Нет» — плечи развилки тоже меняются местами. Следовательно, рокировка — равносильное преобразование алгоритма.

При операции «Да/Нет» дело обстоит иначе, так как все остальные элементы остаются на своих местах. Следовательно, операция «Да/Нет» изменяет алгоритм.

Операция «Да/Нет» изменяет алгоритм, а при рокировке алгоритм остается неизменным

## ВЫВОДЫ

1. Скорость конструирования и вычерчивания медицинских алгоритмов играет важную роль. Чем больше скорость, тем больше производительность труда врачей-разработчиков алгоритмов.
2. Автоматизация разработки медицинских алгоритмов определяется программой ДРАКОН-конструктор.
3. ДРАКОН-конструктор — общее название, обозначающее программы для конструирования и вычерчивания алгоритмов, перечисленные в п. 4.
4. В наличии имеются 4 бесплатные компьютерные программы, поддерживающие разработку медицинских алгоритмов на языке ДРАКОН:
  - «DRAKON Editor Web»,
  - «ИС Дракон»,
  - «Фабула»,
  - «DRAKON Editor».
5. Интернет-консультации по языку ДРАКОН и ДРАКОН-конструктору можно получить на форуме языка ДРАКОН <http://forum.oberoncore.ru/viewforum.php?f=77>
6. Официальный сайт языка ДРАКОН <http://drakon.su/>
7. Видео и презентации [http://drakon.su/video\\_i\\_prezentacii/start](http://drakon.su/video_i_prezentacii/start)
8. Хотя общее число икон и макроикон языка ДРАКОН равно 29, для построения любой дракон-схемы достаточно иметь графическое меню, содержащее всего 20 кубиков.
9. Графическое меню дает возможность конструировать дракон-схему методом «сборки из кубиков». Для этого служит операция «Вставь кубик», позволяющая доставать кубики из меню и укладывать их в проектируемую дракон-схему.
10. Другие операции («пересадка лианы», «заземление лианы» и т. д.) разрешают вносить в схему логические детали, расширяющие ее функциональные возможности и улучшающие эргономическое качество.
11. Во внутренних механизмах ДРАКОН-конструктора реализован полный набор правил языка ДРАКОН, что освобождает пользователя от необходимости запоминать правила.
12. ДРАКОН-конструктор создает только правильно построенные графические схемы и исключает возможность появления запрещенных схем.

**Как связаться с автором?**

Электронная почта: [vdp2007@bk.ru](mailto:vdp2007@bk.ru)  
Тел. 8 (495) 331-50-72  
8 (916) 111-91-57

# СПИСОК ЛИТЕРАТУРЫ

- [1] To Err is Human: Building a Safer Health System / Linda T. Kohn, Janet M. Corrigan, and Molla S. Donaldson, editors. — Committee on Quality of Health Care in America, Institute of Medicine. 2000. — 312p. — ISBN 0-309-06837-1 —, <http://www.nap.edu/catalog/9728.html>.
- [2] Crossing the Quality Chasm: a New Health System for the 21st Century. — Committee on Quality Health Care in America, Institute of Medicine. 2001. — xxi + 337p. — 364 p. —, ISBN: 0-309-51193-3 — <http://www.nap.edu/catalog/10027.html>.
- [3] Improving Diagnosis in Health Care. / Erin P. Balogh, Bryan T. Miller, and John R. Ball, Editors. — Committee on Diagnostic Error in Health Care. Board on Health Care Services. Institute of Medicine., — The National Academies of Sciences, Engineering, and Medicine. — Prepublication copy: Uncorrected proofs. — 450 p. — ISBN 978-0-309-37769-0 — <http://www.nap.edu/21794>.
- [4] Patient Safety and Quality Improvement Act // Wikipedia. [2009—2015]. — [https://en.wikipedia.org/wiki/Patient\\_Safety\\_and\\_Quality\\_Improvement\\_Act](https://en.wikipedia.org/wiki/Patient_Safety_and_Quality_Improvement_Act).
- [5] An Act To amend title IX of the Public Health Service Act to provide for the improvement of patient safety and to reduce the incidence of events that adversely effect patient safety. —, <http://www.gpo.gov/fdsys/pkg/BILLS-109s544enr/pdf/BILLS-109s544enr.pdf>.
- [6] Медицина в афоризмах и крылатых выражениях от истоков до наших дней. / Ачкасов Е.Е., Мискарян И.А., — М.: «Профиль – 2С», 2009. — 448с. — С. 198..
- [7] Паронджанов В. Д. Занимательная информатика, или Волшебный Дракон в гостях у Мурзика. — М.: Росмэн, 2000. — 160 с. — ISBN 5-257-00929-3. —, [http://drakon.su/\\_media/biblioteka\\_1/zanim\\_inf\\_optimizacija1\\_.pdf](http://drakon.su/_media/biblioteka_1/zanim_inf_optimizacija1_.pdf).
- [8] Форумы сайта «Визуальный язык ДРАКОН». — <http://forum.oberoncore.ru/viewforum.php?f=77>.
- [9] Специализированная реанимация новорожденного. Учебник. / Под ред. Р. Й. Надишаускене. — Литва: Центр исследования кризисов, Университет наук здоровья Литвы, 2012. — 396 с. — С. 280–281.
- [10] 1. Цит. по: Роль человеческого фактора в языке. Язык и картина мира. / Б.А. Серебренников, Е.С. Кубрякова, В.И. Постовалова и др. / Под ред. Б.А. Серебренникова. — М.: Наука, 1988. — 216 с. — С. 202. — ISBN 5-02-010880-4..
- [11] Клинические рекомендации. Офтальмология. / под редакцией Л. К. Мошетовой, А. П. Нестерова, Е. А. Егорова. — М.: ГЭОТАР-Медиа, 2008. — 256 с. — С. 109. — ISBN 978-5-9704-0872-8.
- [12] Intravenous fluid therapy. Intravenous fluid therapy in adults in hospital. Clinical Guideline CG174. Methods, evidence and recommendations. — National Clinical Guideline Centre. December 2013. — 194p. —, <http://www.nice.org.uk/guidance/cg174/resources/cg174-intravenous-fluid-therapy-in-adults-in-hospital-guideline2> — См. примеры важных, но неэргономичных графических алгоритмов на стр. 42, 68, 92, 129, 148, 156, Commissioned by the National Institute for Health and Care Excellence. NICE United Kingdom. —.
- [13] UK Ambulance service clinical practice guidelines (2006). / Editors: Joanne D Fisher, Simon N Brown, Matthew W Cooke. — Issued October 2006. — 428p. — ISBN 1-84690-060-3. —,

- http://www2.warwick.ac.uk/fac/med/research/hsri/emergencycare/prehospitalcare/jr calcstakeholderwebsite/guidelines/clinical\_guidelines\_2006.pdf — См. примеры алгоритмов на стр. 139, 141, 146, 147, 152, 298, 338, 396, 401, 409.
- [14] Michael A. Weber at al. Clinical practice guidelines for the management of hypertension in the community. A statement by the American society of hypertension and the International society of hypertension. —, См. пример важного, но неэргономичного графического алгоритма на стр. 7, The Journal of clinical hypertension. Volume 16, Issue 1. (2014). — Article first published online: 17 dec 2013, pp. 1-13. — URL: [http://www.ash-us.org/documents/ash\\_ish-guidelines\\_2013.pdf](http://www.ash-us.org/documents/ash_ish-guidelines_2013.pdf) —.
- [15] Dean S. Hartley. The language of algorithms. — In: Textbook of treatment algorithms in psychopharmacology / edited by Jan Fawcett, Dan J. Stein, and Kenneth O. Jobson. — John Wiley & Sons, 1999. — 205p. —, ISBN 0-471-98109-5. — <https://home.comcast.net/~dshartley3/PSYCHALG/PSYCHALG.HTM>.
- [16] Nurit Barak, Carmi Z. Margolis, Lawrence K. Gottlieb. Text-to-algorithm conversion to facilitate comparison of competing clinical guidelines. — Medical Decision Making. 1998. Volume 18. No 3. July-Sept. pp. 304–310. —, <http://umg.umdnj.edu/smdm/pdf/18-03-304.pdf>.
- [17] Algorithms. Updated consensus guidelines for managing abnormal cervical cancer screening tests and cancer precursors. — American Society for Colposcopy and Cervical Pathology, April 2013. — 24p. —, <http://www.asccp.org/portals/9/docs/algorithms%207.30.13.pdf> .
- [18] Decision making in medicine: an algorithmic approach / edited by Stuart B. Mushlin, Harry L. Greene II. — Third edition. — Mosby Elsevier, 2010 — ISBN 978-0-323-04107-2. — xvi + 726p. —, <http://sgh.org.sa/Portals/0/Articles/Decision%20Making%20in%20Medicine-%20An%20Algorithmic%20Approach.pdf> .
- [19] Texas medication algorithm project. Procedural manual. Bipolar disorder algorithms. / M. Lynn Crismon, Tami R. Argo, Sherrie D. Bendele, Trisha Suppes. — Texas Department of State Health Services. 2007. — 63p. —, <http://www.cardinalinnovations.org/docs/TIMABDman2007.pdf> .
- [20] BMA. Family Doctor Home Adviser. BMA Consulting Medical Editor Dr. Michael Peters. Fifth edition fully revised and updated. — Publisher: Dorling Kindersley Publishers, Ltd. London. 2012. — 296p. — pp. 42–284.
- [21] BMA. When Your Child is Ill: A Home Guide for Parents (BMA Family Doctor) by Bernard Valman (Author). — Publisher: Dorling Kindersley. Third Edition. 2008. — 216p.
- [22] Поликлиническая терапия. Учебник. Под ред. И.Л. Давыдкина, Ю.В. Щукина. — М.: ГЭОТАР-Медиа, 2013. — 688с.
- [23] Практическое руководство для врачей общей (семейной) практики / Под ред. академика РАМН И. Н. Денисова. — М.: ГЭОТАР-МЕД, 2001. — 720 с. — ISBN 5-9231-0050-9.
- [24] Казанцев А.П., Казанцев В.А. Дифференциальная диагностика инфекционных болезней. Руководство для врачей. — М. Изд. Медицинское информационное агентство, 2013. — 496с. — С. 119–463.
- [25] Ваш семейный доктор. Домашний советчик. Перевод с английского. / Предисловие А. И. Воробьева, Н. Е. Шкловского. — М.: Мир, 1992. — 319с — ISBN 5-03-001167-6..

- [26] Теоретические основы языка ДРАКОН // Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. — М.: ДМК Пресс, 2012. — 520с. — С. 425–472. —, ISBN 978-5-94074-800-7 — [http://drakon.su/\\_media/biblioteka\\_1/01..](http://drakon.su/_media/biblioteka_1/01..)
- [27] To Err is Human: Building a Safer Health System / Linda T. Kohn, Janet M. Corrigan, and Molla S. Donaldson, editors. — Committee on Quality of Health Care in America, Institute of Medicine. 2000. — 312p. — Р. 51.
- [28] Чернобай Г.Н., Медицинский диагноз. Лекция, Кемерово: Кемеровская медицинская академия. — 66с. — С. 46. — [http://www.kemsma.ru/counter/Patanatomia\\_Diag.pdf](http://www.kemsma.ru/counter/Patanatomia_Diag.pdf) .
- [29] Медицина в афоризмах и крылатых выражениях от истоков до наших дней. / Ачкасов Е.Е., Мискарян И.А., — М.: «Профиль – 2С», 2009. — 448с. — С. 196.
- [30] Там же. С. 198.
- [31] Там же. С. 366.
- [32] Там же. С. 369.
- [33] Medical errors // Wikipedia. [2004—2015]. [https://en.wikipedia.org/wiki/Medical\\_error](https://en.wikipedia.org/wiki/Medical_error).
- [34] Медицина в афоризмах и крылатых выражениях от истоков до наших дней. — С. 366..
- [35] Там же. С. 369.
- [36] About the Institute of Medicine. <http://iom.nationalacademies.org/About-IOM.aspx>.
- [37] Там же. С. 1, 26, 31.
- [38] Hayward R.A., Hofer T.P. Estimating Hospital Deaths Due to Medical Errors: Preventability Is in the Eye of the Reviewer // JAMA: the Journal of the American Medical Association. — July 25, 2001, Vol. 286, No. 4. — pp. 415–420. —, <http://jama.jamanetwork.com/article.aspx?articleid=194039>.
- [39] To Err is Human. P. 1.
- [40] To Err is Human. P. 42.
- [41] Regional Strategy for Patient Safety in the WHO South-East Asia Region. World Health Organization. Regional Office for South-East Asia. Indraprastha Estate, Mahatma Gandhi Marg, New Delhi. India. 2015. vi + 71p. P. 2, 3.
- [42] Guide for Developing National Patient Safety Policy and Strategic Plan. Patient Safety Unit /Health Systems and Services Cluster. World Health Organization. Regional Office for Africa. Brazzaville. 2014. ix + 35p. P. V, 1–12.
- [43] Восьмой форум по вопросам будущего. Управление безопасностью пациентов. Эрpfендорф, Австрия, 28–29 апреля 2005 г. — Всемирная организация здравоохранения. Европейское региональное бюро. Copenhagen, Denmark, 2006. — 38с. — С. 3.
- [44] Hayward 2001.
- [45] Regional Strategy 2015.
- [46] Guide 2014.
- [47] Восьмой форум 2006.
- [48] Interim Report on the Study of Patient Safety in Maryland. Executive Summary. — Maryland Health Care Commission. 2002. — pp. 71–73. — [http://mhcc.dhmh.maryland.gov/patientsafety/Documents/patient\\_safety/patientsirpt.pdf](http://mhcc.dhmh.maryland.gov/patientsafety/Documents/patient_safety/patientsirpt.pdf).
- [49] Guide 2014. P. 2.

- [50] Sharpe V.A. Promoting Patient Safety. An Ethical Basis for Policy Deliberation. A Special Supplement to the Hastings Center Report. 2003. — 20 p.
- [51] Jewell K., McGiffert L. To Err is Human – To Delay is Deadly. Ten years later, a million lives lost, billions of dollars wasted. — Consumers Union's Safe Patient Project. 2009. — 13p.
- [52] Паронджанов В. Д. Можно ли улучшить медицинский язык? // Человек. 2015, №6. (В печати).
- [53] Гиппократ. Избранные книги. Перевод с греч. проф. В.И. Руднева. — М.: Биомедгиз, 1936. — 736 с. — С. 87.
- [54] World Medical Association. Declaration of Geneva. — <http://www.wma.net/en/30publications/10policies/g1/>.
- [55] Алиев Т. М., Вигдоров Д. И., Кривошеев В. П. Системы отображения информации. — М.: Высшая школа, 1988. — 223с.
- [56] Венда В. Ф. Средства отображения информации. Средства отображения информации. Эргономические исследования и художественное конструирование. — М.: Энергия, 1969. — 303с.
- [57] Венда В. Ф. Инженерная психология и синтез систем отображения информации. — М.: Машиностроение, 1982. — 344с.
- [58] Man-Machine Interface in Nuclear Industry. Conference Proceedings. Tokyo, 15–19 Febr. 1988. — Vienna: International Atomic Energy Agency, 1988.
- [59] Алиев 1988. — С. 10, 57–64.
- [60] Основы инженерной психологии / Б. А. Душков, Б. Ф. Ломов, В. Ф. Рубахин, Б. А. Смирнов. Под ред. Б. Ф. Ломова. — 2-е изд., доп. и перераб. — М.: Высшая школа, 1986. — 448 с.
- [61] Человеческий фактор: В 6 томах. — М.: Мир, 1991, 1992.
- [62] Венда 1969. — С. 144–178.
- [63] Венда 1982. — С. 321–327.
- [64] Венда В. Предисловие к русскому изданию. // Боумен У. Графическое представление информации. — М.: Мир, 1971. — 225с. — С. 5–14.
- [65] Голенков В.В., Гулякина Н.А. Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Часть 1. Принципы создания // Онтология проектирования. — № 1, 2014. — С. 42–64. —, [http://agora.guru.ru/scientific\\_journal/files/Ontology\\_of\\_Designing\\_1\\_2014\\_opt.pdf](http://agora.guru.ru/scientific_journal/files/Ontology_of_Designing_1_2014_opt.pdf).
- [66] Бенвенист Э. Общая лингвистика. — М.: Прогресс, 1974. — 448с. — С. 293.
- [67] Будагов Р.А. Что такое развитие и совершенствование языка? — 2-е изд., доп. — М.: «Добросвет-2000», 2004. — 304с. — ISBN 5-94119-923-9.
- [68] Паронджанов В.Д. Почему мудрец похож на обезьяну, или Парадоксальная энциклопедия современной мудрости. — М.: РИПОЛ Классик, 2007. — 1154с. — С. 231, 240–242, 439–463, 470, 542–544, 663–666, 756, 777, 794. —, [http://drakon.su/knigi\\_vladimira\\_parondzhanova.\\_skachat](http://drakon.su/knigi_vladimira_parondzhanova._skachat).
- [69] Тавровский В.М. Лечебно-диагностический процесс. Теория. Алгоритмы. Автоматизация. — Тюмень: СофтДизайн, 1997. — 320 с. — С. 164, 165.
- [70] Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. — М.: ДМК Пресс, 2014. — 520с. — ISBN 978-5-94074-800-7 —, [http://drakon.su/\\_media/biblioteka\\_1/01.\\_2012\\_uchis\\_chitat\\_new\\_end\\_podlinnik.pdf](http://drakon.su/_media/biblioteka_1/01._2012_uchis_chitat_new_end_podlinnik.pdf).

- [71] Паронджанов В.Д. Дружелюбные алгоритмы, понятные каждому. Как улучшить работу ума без лишних хлопот. — М.: ДМК-пресс, 2010. — 464с. — ISBN 978-5-94074-606-5 —, [http://drakon.su/\\_media/biblioteka\\_1/03.\\_2010\\_druzheljubnye\\_algoritmy\\_1.pdf](http://drakon.su/_media/biblioteka_1/03._2010_druzheljubnye_algoritmy_1.pdf).
- [72] Паронджанов В.Д. Как улучшить работу ума. Алгоритмы без программистов — это очень просто! — М.: Дело, 2001. — 360 с. — ISBN 5-7749-0211-0 —, [http://drakon.su/\\_media/biblioteka\\_1/parondzhanov\\_v.d.\\_kak\\_uluchshit\\_rabotu\\_um\\_a\\_.pdf](http://drakon.su/_media/biblioteka_1/parondzhanov_v.d._kak_uluchshit_rabotu_um_a_.pdf).
- [73] Паронджанов В. Д. Как улучшить работу ума. (Новые средства для образного представления знаний, развития интеллекта и взаимопонимания). — М.: Радио и связь, 1998, 1999. — 352 с. — ISBN 5-256-01211-8.
- [74] Безель Я. Б. Можно ли улучшить работу ума? Новый взгляд на проблему // Вестник РАН. № 4. 2003. Том 73. — С. 363–365. —, (Рецензия на книгу: Паронджанов В.Д. Как улучшить работу ума. Алгоритмы без программистов — это очень просто! — М.: Дело, 2001).
- [75] British Medical Association. Family Doctor Home Adviser. BMA Consulting Medical Editor Dr. Michael Peters. Fifth edition fully revised and updated. — Publisher: Dorling Kindersley Publishers, Ltd. London. 2012. — 296 p. — ISBN 978-1-4093-8334-5.
- [76] British Medical Association. When your child is ill. Dr. Bernard Valman. Third edition. — Publisher: Dorling Kindersley Publishers, Ltd. London. 2008. — 216 p. — ISBN 978-1-4053-2232-4.
- [77] Алгоритмы диагностики и лечения в хирургии / Мак-Интайр Р. Б., Стигманн Г. В., Айсман Б. Перевод с английского под ред. акад. В. Д. Федорова, член-корр. В. А. Кубышкина. — М.: ГЭОТАР-Медиа, 2009. — 744 с. — ISBN 978-5-9704-1236-7.
- [78] Gareth Kantor, John R. Svirbely, Kathy Johnson et al. MEDAL: The Medical Algorithm Project // MEDINFO 2001. Proceedings of the 10th World Congress on Medical Informatics V. Patel et al (eds). — Amsterdam, IOS Press. — Р. 298. — ISBN 1-58603-194-5.
- [79] Цыбулькин Э. К. Неотложная педиатрия. Алгоритмы диагностики и лечения. — М.: ГЭОТАР-Медиа, 2015. — 160 с. — ISBN 978-5-04-3489-5.
- [80] Вялов С.С. Противомикробная терапия: алгоритмы выбора. Практическое руководство. 5-е издание, исправ. и доп. — М.: МЕДпресс-информ, 2015. — 224 с. — ISBN 978-5-00030-264-4.
- [81] Румянцева С. А., Ступин В. А., Афанасьева В. В., Силина Е. В. Алгоритмы и схемы терапии заболеваний, часто встречающихся в терапевтической практике. (Краткое практическое руководство для врачей разных специальностей). — М.-СПб.: Медицинская книга, 2012. —, 432 с. — ISBN 978-5-86093-324-X.
- [82] The Medical Algorithms Company (Medal) is a technology company that provides online access to over 20,000 scientific tools for clinical analytics and medical decision support <http://medicalalgorithms.com/>.
- [83] Ланда Л. Н. Алгоритмизация в обучении. / Под общей ред. и со вступительной статьей Б. В. Гнеденко и Б. В. Бирюкова. — М.: Просвещение, 1966. — 523 с.
- [84] Instructional Development Paradigms / Charles R. Dills, Alexander J. Romiszowski. Editors. — Educational Technology Publications, Englewood Cliff, New Jersey, 1997. — 893 p. — Р. 673. — ISBN 0-87778-294-6. —, <https://books.google.ru/books?id=VD6fsTI2jzIC&pg=PA673&dq=isbn:0877782946+>

- %22semi-algorithmic+prescription%22&hl=en&sa=X&ved=0CBoQ6AEwAGoVChMIupP22-7dyAlVAf1yCh1L3QOM#v=onepage&q=isbn%3A0877782946%20%22semi-algorithmic%20prescription%22&f=true.
- [85] Arden syntax // Wikipedia. [2005—2015]. — [https://en.wikipedia.org/wiki/Arden\\_syntax](https://en.wikipedia.org/wiki/Arden_syntax).
- [86] Health Level 7 // Wikipedia. [2003—2015]. — [https://en.wikipedia.org/wiki/Health\\_Level\\_7](https://en.wikipedia.org/wiki/Health_Level_7).
- [87] Гиппократ. Избранные книги. Перевод с греч. проф. В. И. Руднева. — М.: Биомедгиз, 1936. — 736 с.
- [88] Гиппократ 1936. — С. 624.
- [89] Библия. Новый Завет господа нашего Иисуса Христа. От Иоанна святое благовествование. — Глава 18 : 38..
- [90] Булгаков М. Мастер и Маргарита. М.: Эксмо, 2015. — 640 с. — Часть первая. Глава 2. Понтий Пилат..
- [91] Паронджанов Учись 2014. — С. 154.
- [92] McCarthy Formalism // Wikipedia. [2006—2015]. — [https://en.wikipedia.org/wiki/McCarthy\\_Formalism](https://en.wikipedia.org/wiki/McCarthy_Formalism).
- [93] Антонова С.Г., Васильев В.И., Жарков И.О., Коланькова О.В., Ленский Б.В., Рябинина Н.З., Соловьева В.И. Редакторская подготовка изданий. Учебник. \ Под ред. проф. С.Г. Антоновой. — М.: МГУП, 2002. — 468с. — С. 371. — ISBN 5-8122-0285-0.
- [94] Поликлиническая терапия. Учебник. / под ред. И. Л. Давыдкина, Ю. В. Щукина. — М.: ГЭОТАР-Медиа, 2013. — 688 с. — С. 623. — ISBN 978-5-9704-2396-7.
- [95] Чугров С. В. Научная статья: плод творчества, ремесла или озарения? // Полис. Политические исследования. — 2015. № 3. — С. 160-176. — С. 174.
- [96] Феофанов А. Н., Скопич Т. И. О часто встречающихся ошибках при написании научной статьи. // Наукові праці УкрНДМІ НАН України. Випуск 12 / Під заг. ред. чл.-кор. НАН України А. В. Анциферов. — Донецьк, УкрНДМІ НАН України, 2013. — 390 с.— С. 16.
- [97] Как писать курсовую работу по истории древнего мира. Учебно-методическое пособие для вузов. Составитель Н.П. Писаревский. — Воронеж: Воронежский гос. университет, 2007. — 15 с.
- [98] Клаус Мозер. Психология маркетинга и рекламы / Пер. с немецкого. — Харьков: Изд-во Гуманитарный Центр, 2004. — 380 с. — С. 278. — ISBN 966-8324-02-1.
- [99] Ridpath J.R., Greene S.M., Wiese C.J. PRISM Readability Toolkit. 3rd edition. — Seattle: Group Health Research Institute, 2007. — 79 р. — Р. 8. — [https://www.nhlbi.nih.gov/files/docs/ghchs\\_readability\\_toolkit.pdf](https://www.nhlbi.nih.gov/files/docs/ghchs_readability_toolkit.pdf).
- [100] Scriptio continua // Wikipedia. [2002—2015]. — [https://en.wikipedia.org/wiki/Scriptio\\_continua](https://en.wikipedia.org/wiki/Scriptio_continua).
- [101] Гиппократ 1936. — С. 50.
- [102] Гиппократ 1936. — С. 695.
- [103] Монтень Мишель. Опыты. Том 2. — С. 72.
- [104] Коменский Я. А. Великая дидактика. — СПб: Типография А. М. Котомина, 1875. — Приложение к журналу «Наша Начальная Школа» на 1875 год. — Глава XIV. — С. 71, 72. — Викитеха. — Великая дидактика (Коменский 1875)/Глава XIV.
- [105] Коменский Я. А. Сочинения. Перевод с чешского и латинского / Ин-т философии. — М.: Наука, 1997. — 476 с. — С. 432. — ISBN 5-02-013554-2.

- [106] Коменский 1875. — С. 71.
- [107] Коменский Я. Великая дидактика. — Т. 1. — Смоленск: Гос. учебно-педагогическое изд-во Наркомпроса РСФСР, 1939. — 320 с. — С. 140.
- [108] Коменский 1875. — С. 72.
- [109] Коменский 1875. — С. 73.
- [110] Чижевский Д.И. Ян Амос Коменский и западная философия. // Коменский Я. А. Сочинения. Перевод с чешского и латинского / Ин-т философии. — М.: Наука, 1997. — 476 с. — С. 7, 8. — ISBN 5-02-013554-2.
- [111] Коменский 1997. — С. 133.
- [112] Декарт Р. Избранные произведения. — М.-Л.: Госполитиздат, 1950. — 712 с. — С. 80, 89.
- [113] Коменский 1997. — С. 140.
- [114] Measuring the Impact of Interprofessional Education on Collaborative Practice and Patient Outcomes. — Committee on Measuring the Impact of Interprofessional Education on Collaborative Practice and Patient Outcomes; Board on Global Health. —, Institute of Medicine: The National Academies Press, 150 p. — ISBN 978-0-309-37282-4 — <http://www.nap.edu/21726>.
- [115] Коменский 1997. — С. 134.
- [116] Коменский 1997. — С. 141.
- [117] Коменский 1997. — С. 136, 137.
- [118] Unicode // Wikipedia [2001—2015]. — <https://en.wikipedia.org/wiki/Unicode>.
- [119] List of Unicode characters // Wikipedia [2007—2015]. — [https://en.wikipedia.org/wiki/List\\_of\\_Unicode\\_characters](https://en.wikipedia.org/wiki/List_of_Unicode_characters).
- [120] Паронджанов В. Д. Графический синтаксис языка ДРАКОН. // Программирование. 1995. Т. 3. № 3. — С. 45–62.
- [121] ДРАКОН // Википедия. [2008—2015]. <http://ru.wikipedia.org/?oldid=70108211>.
- [122] Морозов В.В., Трунов Ю.В. и др. Система управления межорбитального космического буксира «Фрегат» // Вестник ФГУП «НПО им. С. А. Лавочкина». — 2014, № 1. — С. 16–25.
- [123] Паронджанов В.Д. Учись писать, читать и понимать алгоритмы. Алгоритмы для правильного мышления. Основы алгоритмизации. — М.: ДМК Пресс, 2014. — 520с. — С. 515.
- [124] Бузель Я. Б. № 4 // Можно ли улучшить работу ума? Новый взгляд на проблему // Вестник РАН. 2003. Т. 73. — С. 363–365. (Рецензия на книгу: Паронджанов В.Д. Как улучшить работу ума. Алгоритмы без программистов — это очень просто! — М.: Дело, 2001)..
- [125] Информация об учебниках представлена на сайте: <http://www.smp.lt/images/naujienos/Ru%20knugos%20KTC%202015.04.30.pdf>.
- [126] Начальная неотложная акушерская помощь. Учебник. / Под ред. Р. Й. Надишаускене. — Литва: Центр исследования кризисов, Университет наук здоровья Литвы, 2012. — 204с.
- [127] Специализированная реанимация новорожденного. Учебник. / Под ред. Р.Й. Надишаускене. — Литва: Центр исследования кризисов, Университет наук здоровья Литвы, 2012. — 396 с.
- [128] Неотложная медицинская помощь. Учебник. / Под ред. Д. Вайткайтиса. — Литва: Центр исследования кризисов, Университет наук здоровья Литвы, 2012. — 265 с.

- [129] Травма. Учебник. / Под ред. Д. Вайткайтиса. — Литва: Центр исследования кризисов, Университет наук здоровья Литвы. — 2012. — 440 с.
- [130] Vileikytė A., Nadišauskienė R.J. et al. Algoritminės „Drakon“ kalbos pritaikymas medicinoje // Lietuvos akušerija ir ginekologija 2014 rugsėjis, tomas XVII, Nr. 3. 192–196.
- [131] Паронджанов В. Д. Неожиданные уроки космонавтики XX века. Новая роль человеческого фактора и когнитивная революция в информационных технологиях / Под ред. Г. Е. Лозино-Лозинского. — М.: Российская инженерная академия, 1995. — Т. 2. Крылатые космические системы. — С. 337–345.
- [132] Паронджанов В.Д. Дружелюбные алгоритмы, понятные каждому. Как улучшить работу ума без лишних хлопот. — М.: ДМК-пресс, 2010. — 464с. — С. 47. — [http://drakon.su/\\_media/biblioteka\\_1/03.\\_2010\\_druzheljubnye\\_algoritmy\\_1.pdf](http://drakon.su/_media/biblioteka_1/03._2010_druzheljubnye_algoritmy_1.pdf).
- [133] Паронджанов В.Д. Как улучшить работу ума. Алгоритмы без программистов — это очень просто! — М.: Дело, 2001. — 360с. — С. 37, 38. — [http://drakon.su/\\_media/biblioteka\\_1/parondzhanov\\_v.d.\\_kak\\_uluchshit\\_rabotu\\_um\\_a\\_.pdf](http://drakon.su/_media/biblioteka_1/parondzhanov_v.d._kak_uluchshit_rabotu_um_a_.pdf).
- [134] Бондарев П.А., Колганов С.К. Основы искусственного интеллекта. — М.: Радио и связь, 1998. — 128с. — С. 65.
- [135] Поликлиническая 2013. — С. 223.
- [136] Поликлиническая 2013. — С. 182, 183.
- [137] Румянцева 2012. — С. 244.
- [138] Поликлиническая 2013. — С. 114-115.
- [139] De Morgan's laws // Wikipedia [2002 – 2016]. — [https://en.wikipedia.org/wiki/De\\_Morgan%27s\\_laws](https://en.wikipedia.org/wiki/De_Morgan%27s_laws).
- [140] Цит. по: Попов Ю. П., Пухначев Ю. В. Математика в образах. — М.: Знание, 1989. — 208 с. — С. 202. — ISBN 5-07-000096-9.
- [141] Неотложная 2012. — С. 87–89.
- [142] Специализированная 2012. — С. 140, 141.
- [143] Практическое 2001. — С. 503.
- [144] Цит. по: Попов 1989. — С. 12.
- [145] Цит. по: Шеврин Л.Н. Об эстетичности математики // Известия Уральского гос. ун-та. — 1995, №4. — С. 25-44. — <http://elar.urfu.ru/bitstream/10995/24220/1/iurp-1995-04-04.pdf>.
- [146] Ильин В. В. Критерии научности знания. — М.: Высшая школа, 1989. — С. 76.
- [147] Поликлиническая 2013. — С. 61.
- [148] Примерная программа дисциплины «Информатика». Издание официальное. — М.: Госкомвуз, 1996. — 21 с. — [http://drakon.su/\\_media/biblioteka/progr\\_drakon.pdf](http://drakon.su/_media/biblioteka/progr_drakon.pdf).
- [149] Примерная 1996. — С. 3, 4, 15, 16.
- [150] Примерная 1996. — С. 15.
- [151] Очков В. Ф., Пухначев Ю. В. 128 советов начинающему программисту. — М.: Энергоатомиздат, 1992. — 256 с. — С. 21. — ISBN 5-283-02535-7.
- [152] Окулова Л. П. Проектирование образовательного процесса в соответствии с требованиями педагогической эргономики. // Вестник. Наука и практика. (29 мая 2012 — 31 мая 2012). — Материалы конференции «Инновации и научные исследования, а также их применение на, практике». Варшава. — <http://xn--e1aaifpcds8ay4h.com.ua/pages/view/730>.

- [153] Корзина М.И., Гурьев А.Т., Лысенко В.А., Сальникова П.Ю., Майоров И.С. Концепция магистерской программы «Информационные технологии в дизайне» в рамках направления подготовки «230400 Информационные системы и технологии» // Проблемы подготовки кадров, (СПОИСУ). — СПб., 2011. — 92с. — С. 48. — ISBN 978-5-905687-44-0 — [http://window.edu.ru/resource/396/76396/files/ikt2011\\_trudi.pdf](http://window.edu.ru/resource/396/76396/files/ikt2011_trudi.pdf), в сфере инфокоммуникационных технологий. — Санкт-Петербургская научно-практическая конференция. 18-20 мая 2011 г. — Санкт-Петербург, Сборник трудов конференции / Санкт-Петербургское Общество информатики, вычислительной техники, систем связи и управления.
- [154] Фокин Ю. Г. Теория и технология обучения: деятельностный подход: учебное пособие для студентов высших учебных заведений. — 3-е изд., испр.. — М.: Издательский центр «Академия», 2008. — 240 с. — С. 233. — ISBN 978-5-7695-5259-5.
- [155] Павлова Н. Ф. Стратегическое планирование развития территориальных социальных образований в схемах. — Екатеринбург: Уральское отделение РАН, 2002. — 133с. — С. 25-33с. — ISBN 5-7691-1400-2.
- [156] Паронджанов 2012. — С. 32.
- [157] Паронджанов 2012. — С. 60.
- [158] Цит. по: Краткий психологический словарь-хрестоматия. Составитель Б. М. Петров. Под ред. проф. К. К. Платонова. — М.: Высшая школа, 1974. — 134 с. — С. 45.
- [159] Начальная 2012. — С. 190.
- [160] Начальная 2012. — С. 178.
- [161] Начальная 2012. — С. 179.
- [162] Поликлиническая 2013. — С. 55.
- [163] Краснов А. Н. Некоторые проблемы формирования клинического мышления у студентов // Традиции и инновации преподавания психиатрии и психологии на различных этапах медицинского образования. Материалы научно-методической конференции. — Чебоксары, 2009. —, 166 с. — С. 25-30. — <http://www.medpsy.ru/library/library014.pdf>.
- [164] Лялин Д. В. Об искусстве. Доклад по философии в МГИАИ, 1988. — [http://zhurnal.lib.ru/l/ljalin\\_d\\_w/art.shtml](http://zhurnal.lib.ru/l/ljalin_d_w/art.shtml).
- [165] Тавровский В. М. Алгоритмы действий врача скорой помощи. — Выпуск 56. 1 ноября 2005 г. Рассуждение пятьдесят третье. — <http://subscribe.ru/archive/science.health.ldp/200511/01001008.html>.
- [166] Тавровский 1997. — С. 93.
- [167] Тавровский В. М. Автоматизация лечебно-диагностического процесса. — Тюмень: Вектор Бук, 2009. — 464 с.
- [168] Тавровский 1997. — С. 153.
- [169] Паронджанов В. Д. Можно ли улучшить медицинский язык? // Человек. №1. 2016. — [http://drakon.su/\\_media/11.\\_mozhno\\_li\\_uluchshit\\_medicinskij\\_jazyk\\_.pdf](http://drakon.su/_media/11._mozhno_li_uluchshit_medicinskij_jazyk_.pdf).
- [170] Паронджанов В. Д. Алгоритмизация медицины и реформа медицинского языка // Евразийский союз ученых (ЕСУ) # 11 (20), 2015 | МЕДИЦИНСКИЕ НАУКИ — С. 144–156. — [http://drakon.su/\\_media/10.\\_algoritmizacija\\_mediciny\\_i\\_reforma\\_medicinskogo\\_jazyka\\_.pdf](http://drakon.su/_media/10._algoritmizacija_mediciny_i_reforma_medicinskogo_jazyka_.pdf).

- [171] Ваш семейный доктор. Домашний советчик. / Перевод с англ. / Предисловие А. И. Воробьева, И. Е. Шкловского. — М.: Мир, 1992. — 319 с. — ISBN 5-03-001167-6.
- [172] MediaWiki // Wikipedia [2003 — 2016]. — <https://en.wikipedia.org/wiki/MediaWiki>.
- [173] Web page // Wikipedia [2001 — 2016] — [https://en.wikipedia.org/wiki/Web\\_page](https://en.wikipedia.org/wiki/Web_page).
- [174] Мир врача. Профессиональный портал. — <https://mirvracha.ru/portal/index>.
- [175] International Classification of Diseases (ICD). — World Health Organization. — <http://www.who.int/classifications/icd/en/>.
- [176] Ратнер Г. Л. Советы молодому хирургу. / Предисловие акад. Е. Вагнера. — Самара: Дом печати, 1991. — 252 с. — Раздел «Характер. Талант. Профессия». — С. 20.
- [177] Определение понятия «сертификация» на сайте Всероссийского научно-исследовательского института сертификации (ОАО ВНИИС). — <http://www.vniis.ru/certification>.
- [178] Автандилов Г.Г., Зайратъянц О.В., Кактурский Л.В. Оформление диагноза. Учебное пособие. — М.: Медицина, 2004. — 304с. — С. 19. — ISBN 5-225-04117-5.
- [179] Неотложная медицинская помощь. Учебник. / Под ред. Д. Вайткайтиса. — Литва: Центр исследования кризисов, Университет наук здоровья Литвы. — 2012. — 265 с. — С.140.
- [180] Непейвода Н.Н. Алгоритм // Новая философская энциклопедия: В 4-х т. / Ин-т философии РАН. / Под. ред Степина В.С., Гусейнова А.А. и др. — М.: Мысль, 2010. — Том 1. — 744с. — С. 76. <http://iph.ras.ru/elib/0111.html>.
- [181] Бачиашвили В. Сообщение на форуме «Язык Дракон. Медицинский вариант. Иконы и Макроиконы». — <http://forum.oberoncore.ru/viewtopic.php?p=85575#p85575>.
- [182] Радченко О. Р. Графические средства системного анализа в практике врача-гигиениста. Методическое пособие для врачей и интернов медико-профилактического профиля. — Казань: Казанская гос. медицинская академия, 2006. — 55 с. —, [http://www.kgmu.kcn.ru/sites/default/files/SYST\\_AN.pdf](http://www.kgmu.kcn.ru/sites/default/files/SYST_AN.pdf).
- [183] Ланда Л. Н. Алгоритмизация в обучении. / Под общей ред. и со вступительной статьей Б. В. Гнеденко и Б. В. Бирюкова. — М.: Просвещение, 1966. — 523 с. — Глава 2. § 1. Понятие предписания алгоритмического типа. — С. 35.
- [184] Паронджанов 2000. — Аннотация. — С. 2..
- [185] Люсов Е.А., Колпаков Е.В. Аритмии сердца. Теоретические и хирургические аспекты. М.: ГЭОТАР-Медиа, 2009. — 400 с. — С. 194. — ISBN 978-5-9704-1032-5.
- [186] Palaeography // Wikipedia. [2001—2015]. — <https://en.wikipedia.org/wiki/Palaeography>.
- [187] Гиппократ // Новая философская энциклопедия. В 4-х томах. Том 1. — Ин-т философии РАН, Нац. общ.-научн. фонд. Под ред. В.С. Степина, А.А. Гусейнова. — М.: Мысль, 2010. — 744 с. — С. 531. — ISBN 978-2-244-01116-6.
- [188] Н. Бурцева, Е. Петров. Жирограф и ДРАКОН Пилюгина. Телерадиостудия Роскосмоса. (17 мая 2008). — Фильм выпущен к 100-летию со дня рождения Главного конструктора систем управления ракет-носителей, академика Н. А. Пилюгина. — <http://www.tvroscosmos.ru/3559/>.
- [189] Специализированная 2012. — С. 282.
- [190] Специализированная 2012. — С. 128.

- [191] Неотложная 2012. — С. 137.
- [192] Поликлиническая 2013. — С. 52–73..
- [193] Поликлиническая 2013. — С. 61, 63..
- [194] Н. Г. Коновалова, Куртигешева А. А., Урбанский А. С. Диагностические алгоритмы в контроле клинических умений у студентов медицинского колледжа. // Вестник Кузбасской государственной педагогической академии. Электронный журнал. — 1 (15), январь 2012. —, <http://vestnik.kuzspa.ru/articles/66/>.
- [195] Паронджанов 2014. — С. 153-183.
- [196] Травма 2012. — С. 403.
- [197] Начальная 2012. — С. 79.
- [198] Неотложная 2012. — С. 45.
- [199] Ломов Б. Ф. Эргономические (инженерно-психологические) факторы художественного конструирования // Учебно-методические материалы по художественному конструированию. — Изд. ВХПУ, 1965.
- [200] Поликлиническая 2013. — С. 1.