

**Владимир  
ПАРОНДЖАНОВ**

**ДРУЖЕЛЮБНЫЕ  
АЛГОРИТМЫ,  
ПОНЯТНЫЕ  
КАЖДОМУ**

**КАК УЛУЧШИТЬ РАБОТУ УМА  
БЕЗ ЛИШНИХ ХЛОПОТ**



**Москва  
2010**

# СОДЕРЖАНИЕ

<b>Введение.....</b>	<b>5</b>
<b>Часть I. Зачем нужен дружелюбный алгоритмический язык?.....</b>	<b>17</b>
Глава 1. Интеллектуальный терроризм: фантазия или реальность?.....	19
Глава 2. Психологический подход к алгоритмам.....	36
Глава 3. Можно ли создать алгоритмический язык, улучшающий понимание и взаимопонимание?.....	42
<b>Часть II. Знакомьтесь — дружелюбный алгоритмический язык         ДРАКОН.....</b>	<b>49</b>
Глава 4. Изюминки языка ДРАКОН.....	51
Глава 5. Эргономичные алгоритмы.....	83
Глава 6. Визуализация циклических алгоритмов.....	129
Глава 7. Визуализация логических формул.....	158
Глава 8. Что такое эргономичный текст?.....	178
Глава 9. Визуальные операторы реального времени.....	197
Глава 10. Дружелюбное программирование.....	210
<b>Часть III. Примеры алгоритмов для различных отраслей знания .....</b>	<b>223</b>
Глава 11. Алгоритмы в медицине.....	225
Глава 12. Алгоритмы в промышленности.....	233
Глава 13. Алгоритмы в торговле.....	243
Глава 14. Алгоритмы в атомной энергетике.....	254
Глава 15. Алгоритмы в биологии.....	269
Глава 16. Алгоритмы в сельском хозяйстве.....	279
Глава 17. Алгоритмы в средней школе.....	285
Глава 18. Алгоритмы и задачи государственного, муниципального и корпоративного управления.....	296
Глава 19. Алгоритмы и формализация профессиональных знаний.....	303

Глава 20. Рекомендации по использованию алгоритмических структур «силуэт» и «примитив».....	321
<b>Часть IV. Визуальный конструктор алгоритмов и формальное описание языка ДРАКОН.....</b>	<b>325</b>
Глава 21. Визуальный конструктор алгоритмов (дракон-редактор).....	327
Глава 22. Визуальный синтаксис языка ДРАКОН.....	339
<b>Часть V. Теоретическое обоснование языка ДРАКОН.....</b>	<b>349</b>
Глава 23. Визуальное структурное программирование.....	351
Глава 24. Исчисление икон.....	380
Глава 25. Что день грядущий нам готовит? (Проект преобразования алгоритмического языка).....	391
<b>Часть VI. Возможна ли эргономизация математики? История и эргономика математики и создание языка ДРАКОН.....</b>	<b>405</b>
Глава 26. О чем говорят уроки истории? Анализ эргономичности алгебры Диофанта.....	407
Глава 27. Эргономизация математики. Обсуждение и примеры.....	425
<b>Новое платье алгоритмов: взгляд в будущее (вместо заключения).....</b>	<b>447</b>
<b>Литература.....</b>	<b>452</b>
<b>Предметный указатель.....</b>	<b>461</b>

---

# ВВЕДЕНИЕ

У каждого человека имеются огромные интеллектуальные ресурсы, из которых большинство людей использует лишь незначительную часть.

*Давид Лассер*

## АЛГОРИТМЫ И ЧЕЛОВЕЧЕСКИЙ РАЗУМ

Мы живем в мире алгоритмов, но знаем о них удивительно мало. Многие люди всю жизнь пользуются алгоритмами, не догадываясь об этом. Между тем алгоритмы играют огромную роль в жизни общества. Они оказывают заметное влияние на эффективность экономики и уровень жизни.

К сожалению, многие алгоритмы и программы похожи на загадочный ребус. Они непонятны никому, кроме горстки их создателей. Непонимание порождает путаницу и досадные ошибки. Чтобы поправить дело, надо сделать алгоритмы «дружелюбными». Это позволит превратить алгоритмы-головоломки в наглядные алгоритмы-картинки, обеспечивающие *быстрое и глубокое понимание*. Глубина понимания сложных проблем — как раз то, чего всем нам (от студента до министра) ой как не хватает!

Дружелюбные алгоритмы пишут на **эргономичных** графических языках. Они создают повышенный интеллектуальный комфорт, улучшают работу ума, повышают продуктивность труда. С их помощью вы научитесь легко и быстро, затратив минимум усилий, решать сложнейшие проблемы. Проектировать сложную деятельность и бизнес-процессы. Формализовать свои профессиональные знания. Выполнять алгоритмизацию самостоятельно, без помощи программистов — по методу «программирование без программистов».

## МАЛЕНЬКАЯ УВЕРТЮРА

Книга предназначена для всех, кто хочет упорядочить и улучшить работу своего ума. Она адресована к работникам умственного труда всех профессий и специальностей: конструкторам и педагогам, технологам и врачам, агро-

номам и математикам, биологам и экономистам, психологам и нефтяникам, физикам и программистам и т. д.

Она может пригодиться тем, кто хочет научиться рисовать свои мысли и планы в виде наглядных и точных блок-схем. И разобраться, наконец, что же такое алгоритмы и почему они играют такую важную роль в развитии цивилизации и человеческого интеллекта.

У того, кто бегло пролистал книгу, может сложиться ложное впечатление, что она посвящена компьютерам и программированию. На самом деле это не так. Речь идет не о думающих машинах, а о думающих людях, о загадках и особенностях человеческого познания и интеллекта. О таинственных ловушках и подводных камнях, которые подстерегают нас в трудном плавании по безбрежным морям каждодневной умственной работы. О том, как найти спасительный маяк и прибыть к цели по кратчайшему маршруту.

В книге предлагается новое средство для облегчения и улучшения работы ума, которое можно использовать во многих, практически в любых областях умственной деятельности. Само по себе это средство не имеет никакого отношения к компьютерам. Поэтому его с успехом могут применять и те, кто не любит компьютеры, относится к ним с подозрением и опаской. Вам понадобятся карандаш, бумага и больше ничего. Короче говоря, это средство вполне пригодно для улучшения самой обычной (бескомпьютерной) умственной работы.

Впрочем, любители компьютеров выиграют еще больше, поскольку указанное средство может служить основой для создания новой мощной информационной технологии, которая принесет большую пользу и сейчас и в будущем.

Возможно, книга попадет в руки читателя, который хотел бы улучшить работу своего ума, но которого пугают или раздражают такие слова, как «алгоритм», «программа», «формализация». Этому горю нетрудно помочь. Сейчас мы сочиним шуточный словарь, который хотя и нарушает все каноны научной строгости, зато вполне понятен новичкам.

#### Легкомысленный словарь

**Алгоритм** — точное описание решения задачи, которое ведет к победе

**Алгоритм** — точно описанная последовательность человеческих действий

**Алгоритм** — точное и полное описание работы (деятельности), которое позволяет другим людям повторить эту деятельность фотографически точно, без малейших отклонений, и получить нужный результат

**Визуальный алгоритм** — алгоритм, изображенный не в виде текста, а в виде наглядной картинки



**Визуализация алгоритма** — преобразование алгоритма, который записан в виде плохого и непонятного текста, в хорошую и понятную картинку

**Когнитивный** — познавательный. Это неуклюжее словечко надо запомнить, так как оно будет попадаться на каждом шагу

**Когнитивная психология** — психология познавательных процессов

**Программа** — последовательность действий, которые человек ленился выполнять сам и поэтому поручает компьютеру или роботу

**Формальный** — математически строгий

**Формальное описание** — математически строгое, точное, однозначное и полное описание, лишенное пробелов и двусмысленностей

**Формализация** — превращение обычного (плохого) описания в формальное (хорошее)

**Автоформализация** — это когда человек выполняет формализацию сам, не обращаясь к помощи друзей, родственников и случайных прохожих

**Формализация деятельности** — точное описание правил, по которым выполняется деятельность. В ходе формализации необходимо разбить деятельность на отдельные действия, указать последовательность их выполнения, а также условия, при которых выполняется (или не выполняется) каждое действие. В результате формализации описание деятельности превращается в алгоритм

**Алгоритмизация** — то же самое, что формализация деятельности

**Алгоритмизация** — внесение порядка в царство анархии, устранение путаницы и разгильдяйства, наведение технологической дисциплины

**Алгоритмизация** — процесс создания алгоритма

**Эргономика** — наука о том, как превратить сложную, трудную и противную работу в простую, легкую и приятную

**Когнитивная эргономика** — наука о том, как облегчить и улучшить умственную работу

**Эргономичный** — дружелюбный

**Эргономичный** алгоритм — дружелюбный алгоритм

**Эргономичный** алгоритмический язык — дружелюбный и удобный в работе алгоритмический язык

**Эргономизация алгоритма** — превращение недружелюбного алгоритма в дружелюбный

**Эргономизация алгоритмического языка** — превращение недружелюбного алгоритмического языка в дружелюбный

**Эргономизация образования** — облегчение и улучшение учебной деятельности

## ТРЕТИЙ ГЛАЗ ДЛЯ БИЗНЕСМЕНОВ И РУКОВОДИТЕЛЕЙ

Предположим, вы — крупный руководитель. Например, генеральный конструктор ракетно-космической корпорации. Или даже министр. Или, скажем, глава крупного банка, в котором несколько сотен мощных компьютеров перемалывают финансовую и иную информацию. Или, предположим, вы — главный инженер большого металлургического завода, где успешно действуют самые современные системы управления технологическими процессами. Возможно, вы возглавляете нефтяную компанию, железную дорогу или центр спутникового телевидения.

Гордость вашей организации, ее интеллектуальный костяк составляют золотые умы — квалифицированные специалисты, обладающие драгоценными профессиональными знаниями. В своей работе они используют компьютеры, объединенные в локальные и иные сети. В этой книге описан **практически полезный метод, позволяющий улучшить работу ума этих людей, чтобы увеличить их интеллектуальный вклад в процветание вашей частной фирмы или государственной организации.**

Речь идет о совершенно новой идее, которая, впрочем, уже прошла тщательную проверку в ряде частных случаев и показала хорошие результаты. Предлагаемая идея тесно связана с компьютерами и автоматизацией, но в то же время существенно отличается от них, так как объектом воздействия является не компьютер, а человеческий мозг.

Здесь уместны некоторые пояснения общего характера. Когда говорят об автоматизации, имеют в виду автоматизированные системы управления предприятиями, технологическими процессами, научными исследованиями, конструкторскими разработками, проектированием, программированием, финансовой деятельностью, войсками и множеством других. В XX в. широкая волна компьютеризации и автоматизации охватила весь мир и принесла замечательные плоды. Общественное богатство увеличилось. Доля физического труда сократилась, умственного — возросла. В этот период были осознаны две важные истины.

1. Современный мир — продукт мысли и ума. В конечном счете, именно человеческий ум произвел все то, что мы видим и ощущаем вокруг себя. Цивилизация — это результат усилий человеческого ума.
2. В конкурентном мире успех деятельности фирм и организаций зависит от профессиональных знаний и интеллекта специалистов, от интеллектуального потенциала фирмы, от умения увеличить силу ума работников. Улучшение работы ума специалистов превращается в важнейшую, приоритетную задачу.

Здесь, однако, возникает проблема. Многие считают, что эта задача решается автоматически, сама по себе, вместе с улучшением образования и широким распространением компьютеров. Они полагают, что автоматизация умственного труда усиливает человеческий интеллект. По их мнению, чем совершеннее компьютеры, тем лучше их характеристики,

тем продуктивнее работает мозг. Чем больше сложных и интеллектуальных задач возлагается на машины, тем эффективнее решает свои задачи человеческий интеллект. Чем выше степень автоматизации умственного труда, тем лучше и эффективнее действует наш ум. Согласно этой логике, чтобы улучшить работу ума специалистов, нужно увеличить мощность компьютеров и сделать их более интеллектуальными.

Можно показать, что в этих рассуждениях скрыта тонкая и коварная ошибка. На самом деле автоматизация и улучшение работы ума — хотя и связанные, но *существенно разные вещи*. Творческая продуктивность человеческого мозга не зависит от мощности и других характеристик компьютеров. Она определяется совсем другими факторами.

Таким образом, налицо противоречие. С одной стороны, очевидно, что эффективность фирм и организаций зависит от силы ума специалистов. Поэтому повышение творческой силы интеллекта становится задачей первостепенной важности. С другой стороны, в настоящее время, насколько нам известно, отсутствуют эффективные методы интенсификации человеческого интеллекта, которые можно применить на практике с целью увеличения реальной отдачи фирм и организаций. Более того, задача улучшения работы ума специалистов не только не решена, но фактически даже не поставлена.

Данная книга, по-видимому, представляет собой первую в мировой литературе попытку четко сформулировать проблему улучшения работы ума, выявить ее тесную связь с алгоритмами, которые в огромном количестве населяют всемирную армию компьютеров и головы людей. Показать огромное влияние алгоритмов на человеческий разум. И продемонстрировать неуклонно растущую значимость проблемы «алгоритмы + разум» для повышения эффективности фирм и организаций.

## **НЕУДАЧНЫЕ ПРИКЛЮЧЕНИЯ АЛГОРИТМОВ В ОБЩЕСТВЕ ЗНАНИЙ**

Мы живем в обществе знаний. Носителями профессиональных знаний являются специалисты, например, агрономы, нефтяники, микробиологи, технологи, экономисты, химики, врачи. Большинство указанных специалистов, как правило, далеки от программирования.

Парадокс в том, что эти люди прекрасно знают свою работу. Они четко знают последовательность выполняемых ими действий, то есть алгоритмы своей работы. Но они, увы, не знают, что их знания называются алгоритмами. Эти замечательные специалисты не умеют описывать алгоритмы. Они не умеют расчленять свои знания и выделять среди них алгоритмическую (процедурную) часть. Они не в состоянии выразить свои знания на бумаге в форме алгоритмов.

Хорошо ли это? Допустима ли подобная алгоритмическая неосведомленность?

## УСТАРЕВШАЯ ТОЧКА ЗРЕНИЯ

На этот вопрос обычно отвечают так:

Конечно, хорошо. Потому что существует разделение труда. Каждый должен заниматься своим делом. Алгоритмы — дело математиков и программистов. А нефтяники, химики, врачи и экономисты совершенно не обязаны знать алгоритмические тонкости. У них, слава богу, своих забот выше крыши!

Данная точка зрения требует непредвзятого анализа. Прежде всего, надо ответить на вопрос: В чем причина повсеместно распространенной алгоритмической неграмотности? Неграмотности подавляющего большинства специалистов?

Ответ лежит на поверхности. Дело в том, что нынешние алгоритмы (используемые во всем мире) имеют серьезный дефект. Они *чрезвычайно трудны для понимания*.

Поэтому существующая практика изучения, разработки и эксплуатации алгоритмов является неудовлетворительной. Она требует неоправданно больших затрат труда и времени. Эти трудозатраты настолько велики, что во много раз превышают реальные резервы времени, которыми располагают люди.

Алгоритмическая неграмотность специалистов-непрограммистов объясняется тем, что в современных условиях изучение алгоритмов является слишком сложным и даже непосильным делом. Поэтому работа с алгоритмами для подавляющего большинства профессионалов оказывается невозможной.

Это плохо. Алгоритмическая неграмотность многих умных людей неблагоприятно отражается на развитии общества.

## АЛГОРИТМИЗАЦИЯ ТРУДНА. МОЖНО ЛИ СДЕЛАТЬ ЕЕ ЛЕГКОЙ?

Повторим главную мысль. Нынешние алгоритмы, как правило, недружелюбны. Это обстоятельство ставит непреодолимый барьер и не позволяет перейти к массовому овладению алгоритмизацией.

К счастью, ситуацию можно изменить. Алгоритмы можно и нужно сделать дружелюбными. Как только это случится, ситуация в корне изменится.

Ниже мы предложим более удобную форму записи алгоритмов (но не данных).

## АЛГОРИТМИЗАЦИЯ — ЭТО ТРУД

Чтобы прояснить проблему, сделаем несколько замечаний.

- Разработка алгоритмов — это труд, производительность которого

играет важную роль. Если труд слишком сложен (производительность труда мала), то алгоритмизацию могут выполнять только опытные специалисты. При таких условиях массовая алгоритмизация невозможна. И наоборот, если данный труд удастся резко облегчить, алгоритмизация станет посильной почти для каждого. В этом случае создаются необходимые предпосылки для массового овладения методами алгоритмизации.

- В этой книге мы покажем, что использование дружелюбных алгоритмических языков кардинальным образом облегчает труд и повышает его производительность. Следовательно, дружелюбные языки пригодны для эффективной формализации процедурных знаний, осуществляемой в массовых масштабах.

## ЦЕЛЬ КНИГИ

Наша цель — научить читателя самостоятельно создавать дружелюбные алгоритмы. И показать, что это простое и легкое дело. На многочисленных примерах читатель убедится, что дружелюбные алгоритмы имеют огромные преимущества.

Чем понятнее алгоритм, тем легче уяснить его смысл. Чем прозрачнее смысл, тем лучше взаимопонимание между заказчиком и исполнителем работ. Чем меньше ошибок в техническом задании, тем выше производительность труда при разработке и отладке алгоритмов и программ. Чем меньше усилий затрачивают алгоритмисты и программисты, тем быстрее они выполняют свою работу.

## ЕЩЕ РАЗ ОБ УСИЛЕНИИ ИНТЕЛЛЕКТА

Академик Андрей Ершов был убежден: занятия программированием делают людей умнее. Человек резко увеличит свой интеллект, если станет программистом. Вот его слова:

«Человек неизмеримо усилит свой интеллект, если сделает частью своей природы способность планировать свои действия, вырабатывать общие правила и способ их применения к конкретной ситуации, организовывать эти правила в осознанную и выразимую структуру, — одним словом, сделается программистом» [1].

## НЕДОСТАТОК ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

Прав ли уважаемый академик? Нет сомнения, умение хорошо программировать значительно расширяет умственные возможности людей. Оно позволяет оптимизировать работу ума и использовать компьютер на порядок эффективнее. Однако есть одно «но». На практике предложенный Ершовым путь улучшения интеллекта — увы! — не работает. Он просто-

напросто закрыт для подавляющего большинства желающих. Почему?

Неприятность в том, что все без исключения известные языки программирования слишком сложны и недоступны для большинства людей. Они предназначены для узкой касты знатоков и больше ни для кого. Интеллектуальные трудозатраты на освоение программирования недопустимо велики.

Поэтому языки программирования непригодны для решения задачи, поставленной академиком Ершовым — усиления интеллекта значительной части населения.

## КАК ВЫЙТИ ИЗ ТУПИКА?

Наша позиция такова:

- Надо глубоко осознать разницу между понятиями «программирование» и «алгоритмизация».
- Массовое обучение программированию невозможно и не нужно по двум причинам. Во-первых, оно невероятно трудно. Во-вторых, оно дает знания, которые большинству просто не нужны.
- Массовое обучение алгоритмизации, наоборот, полезно и необходимо.
- В обществе знаний во многих случаях возникает острая необходимость формализовать собственные процедурные профессиональные знания специалистов.
- Такое умение должно стать частью их профессиональной культуры.

## РАБОЧАЯ ГИПОТЕЗА:

### КАЧЕСТВО ЯЗЫКА ВЛИЯЕТ НА СИЛУ УМА

Эргономическое качество алгоритмических языков имеет большое значение. Если качество низкое, алгоритмы оказываются слишком сложными и трудными для понимания. Они содержат смысловые пробелы, упущения, двусмысленности и непонятные места. Это порождает многочисленные неясности, заблуждения, ошибки и взаимное непонимание между исполнителями работ. В итоге производительность умственного труда создателей алгоритмов падает. Но это не все. Чрезмерная трудность понимания алгоритмов ставит непреодолимую стену перед всеми, кто хочет освоить алгоритмизацию.

И наоборот, если качество языка высокое, алгоритмы становятся легкими для восприятия. Растет глубина и скорость понимания и взаимопонимания. Смысловые ошибки, противоречия и слабые места «сами бросаются в глаза». Их можно гораздо быстрее обнаружить и устранить. В результате продуктивность мозга существенно возрастает.

Сказанное позволяет выдвинуть гипотезу.

### Гипотеза

Производительность мозга разработчика алгоритмов зависит от эргономичности алгоритмического языка. Улучшая язык, можно поднять производительность.

Исходя из этих соображений был разработан язык ДРАКОН, который описан ниже.

Язык ДРАКОН обладает уникальными эргономическими характеристиками. Он позволяет улучшить работу ума, легко и быстро создавать дружелюбные и наглядные алгоритмы.

Мы попытаемся доказать, что язык ДРАКОН — мощное оружие интеллекта. Он позволяет ясно и четко мыслить. И значительно облегчает творческий процесс создания алгоритмов, делая его доступным не только для программистов, но и для «народа».

## ДРАКОН РОДИЛСЯ В КОСМИЧЕСКОЙ КОЛЫБЕЛИ

Язык ДРАКОН разработан совместными усилиями Федерального космического агентства (Научно-производственный центр автоматики и приборостроения им. акад. Н.А. Пилюгина, г. Москва) и Российской академии наук (Институт прикладной математики им. акад М.В. Келдыша, г. Москва).

ДРАКОН возник как обобщение опыта работ по созданию космического корабля «Буран». Разработка нового языка и системы программирования началась в 1986 и завершилась через 11 лет. В 1996 году на базе ДРАКОНа построена автоматизированная технология проектирования алгоритмов и программ под названием «ГРАФИТ-ФЛОКС». Она успешно используется во многих крупных космических программах, включая:

- международный космический проект «Морской старт» (Sea Launch); Первый пуск «Морского старта» состоялся 10 марта 1999 года;
- разгонный блок космических аппаратов «Фрегат»;
- модернизированная ракета-носитель тяжелого класса «Протон-М» и др.

Поскольку результаты использования ДРАКОНа были стабильно высокими, руководство Пилюгинского центра приняло решение об использовании дракон-технологии во всех последующих проектах.

## ПРОГРАММИРОВАНИЕ БЕЗ ПРОГРАММИСТОВ

ДРАКОН — очень легкий язык. Настолько легкий, что разработку многих компьютерных программ для космических ракет на практике ведут не программисты, а инженеры — по принципу «программирование без программистов». Причина частичного отказа от программистов проста. При

решении практических прикладных задач инженеры досконально владеют материалом и прекрасно знают постановку задачи. В отличие от них программисты не знают физику процесса и становятся «лишними людьми», без которых в ряде случаев (хотя и не всегда) вполне можно обойтись.

Это позволяет значительно сократить издержки, улучшить показатель «затраты — результат», ускорить ход работ. И полностью избавиться от ошибок «испорченного телефона», вызванных взаимным непониманием между программистами и инженерами.

## СЕКРЕТЫ КОСМИЧЕСКИХ РАЗРАБОТОК — НАРОДНОМУ ХОЗЯЙСТВУ

ДРАКОН универсален. Он может применяться для наглядного представления и быстрой разработки алгоритмов не только в «космосе», но и в «земных» видах человеческой деятельности. Практическая полезность ДРАКОНа получила высокую оценку.

Можно предположить, что язык ДРАКОН получит широкое распространение в самых различных областях, в том числе в системе образования.

В свое время Никлаус Вирт, автор языка Паскаль, посчитал, что Паскаль должен быть самым первым языком, с которого следует начинать изучение программирования. Эта точка зрения в свое время стала почти общепринятой.

В ту пору программы писали в виде *текста*. Для *текстового* программирования Паскаль, действительно, был наилучшим учебным языком.

Однако сегодня ситуация изменилась. Будущее принадлежит эргономичным языкам. В этих условиях дедушка Паскаль потерял былую славу прекрасного учебного средства.

Сегодня эта роль переходит к визуальному языку ДРАКОН. Именно ДРАКОН становится самым простым, легким, удобным и логически стройным языком, с которого надо начинать изучение алгоритмизации.

## ДРАКОН В СИСТЕМЕ ОБРАЗОВАНИЯ

В 1996 году Государственный комитет по высшему образованию Российской Федерации включил изучение языка ДРАКОН в программу курса информатики высшей школы. Этот факт нашел отражение в официальном документе Госкомвуза под названием:

«Примерная программа дисциплины «Информатика».  
Для направлений:  
510000 — Естественные науки и математика  
540000 — Образование  
550000 — Технические науки



560000 — Сельскохозяйственные науки

Издание официальное.

М.: Госкомвуз, 1996. 21с.

Авторы программы: Кузнецов В.С., Падалко С.Н., Паронджанов В.Д., Ульянов С.А.

Научные редакторы: Падалко С.Н., Паронджанов В.Д.» [2].

*(Примечание. Этот документ официально «узаконил» изучение языка ДРАКОН в системе образования).*

В настоящее время ведется подготовка учебных книг для средней и высшей школы. Уже издана первая из них — учебное пособие по алгоритмам для учащихся 5 — 9 классов. Это пособие выдержало три издания [3—5].



## СОДЕРЖАНИЕ КНИГИ

Книга состоит из шести частей.

Часть I (главы 1—3) носит вводный характер. Практика показывает, что многие алгоритмисты и программисты, в том числе опытные, с недоверием воспринимают эргономический подход к алгоритмам. Они привыкли думать, что теоретической основой программирования является математика, но никак не эргономика. Мысль о том, что математика должна вступить в союз с эргономикой кажется им крамольной и еретической. Первая часть книги призвана развеять эти сомнения.

Сделаем оговорку: читатели, которые не нуждаются в подобных разъяснениях, могут без всякого ущерба пропустить первую часть и начать чтение с 4-й главы.

В части II (главы 4–10) описывается эргономичный алгоритмический язык ДРАКОН. Для удобства читателя описание языка дается в упрощенном виде, на наглядных примерах.

Часть III (главы 11–20) содержит большое число алгоритмов на языке ДРАКОН. Примеры демонстрируют широкий спектр возможностей языка, позволяющих использовать его для различных отраслей и предметных областей.

В части IV (главы 21, 22) описывается программа «конструктор алгоритмов» и дается формальное описание языка.

Часть V (главы 23–25) посвящена теоретическому обоснованию языка ДРАКОН. В главе 23 описана теория визуального структурного программирования. (Именно она лежит в основе ДРАКОНа). В главе 24 показано, что визуальный синтаксис ДРАКОНа представляет собой визуальное логическое исчисление, которое называется *исчислением икон*. В главе 25 представлен Проект преобразования алгоритмического языка.

Часть VI (главы 26, 27) освещает историю дружеских связей математики и эргономики. Характерная для ДРАКОНа «дружба» математики и эргономики вовсе не является исключением. История математики на многих фактах подтверждает, что подобная «дружба» является одной из важных закономерностей развития науки.