

An introduction to

DRAKON

Stepan Mitkin

Who invented DRAKON?

Original idea

Edsger Wybe Dijkstra

Final look

Russian Space Program:
Vladimir Parondzhanov

Why DRAKON?

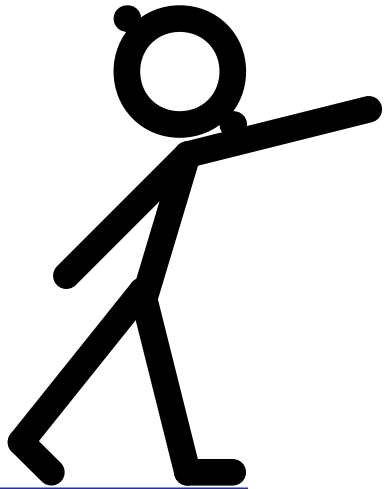


Spacecraft control systems

Ultra-high complexity

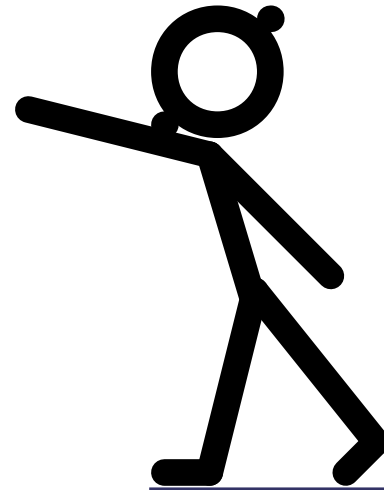
Ultra-high cost of failure

The bottleneck



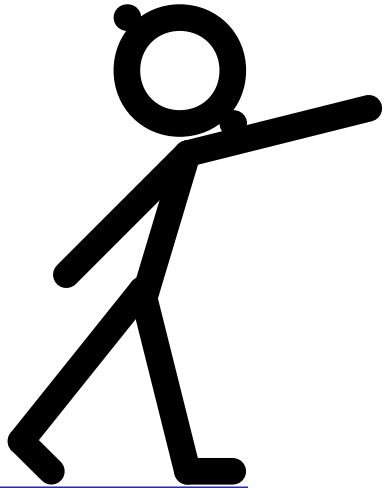
Engineers

?



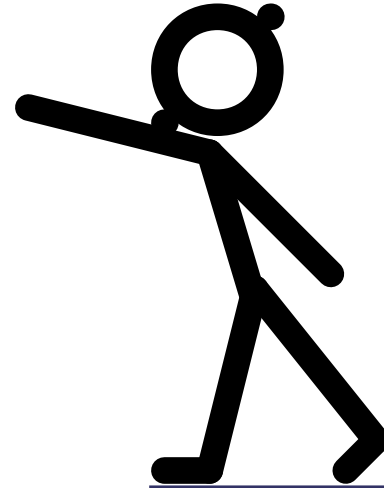
Programmers

The bottleneck



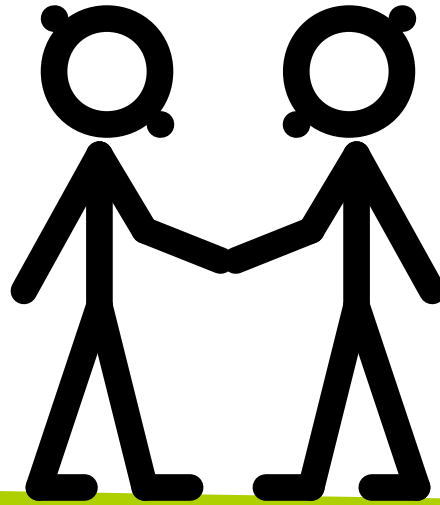
WHAT
needs to
be done?

?



HOW to
implement
that?

The solution



DRAKON

WHAT
needs to
be done?

HOW to
implement
that?

What is DRAKON?

visual

algorithmic

language

Algorithm

How to

```
graph TD; A[How to] --- B[Reach a goal]; A --- C[Behave];
```

Reach a goal

Behave

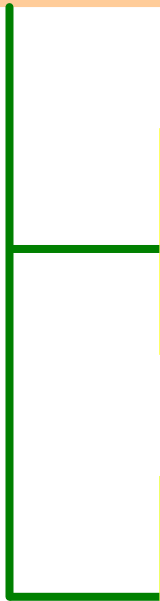
Algorithm

How to

Unambiguous

Reach a goal

Behave



Text information

A view down a fjord.

Camera elevation: 750 m.

Steep mountains on the left side. Height: approx. 700 m.

Moderate steep mountains on the right side. Height:
approx. 650 m.

Weather: sunny, partially cloudy.

Flat-topped rock with vertical walls.

Some people.

Visual information



Language

how to use symbols

how NOT to use symbols

Languages similar to DRAKON

- Flowcharts
- Activity diagrams (UML)
- Decision trees
- State machine diagrams (UML)

Why is DRAKON special?

Highly tuned rules

Unique features

Highly tuned rules

rectangular planar graph

reduced visual noise

no line intersections

many others

Rectangular planar graph

allows

bans

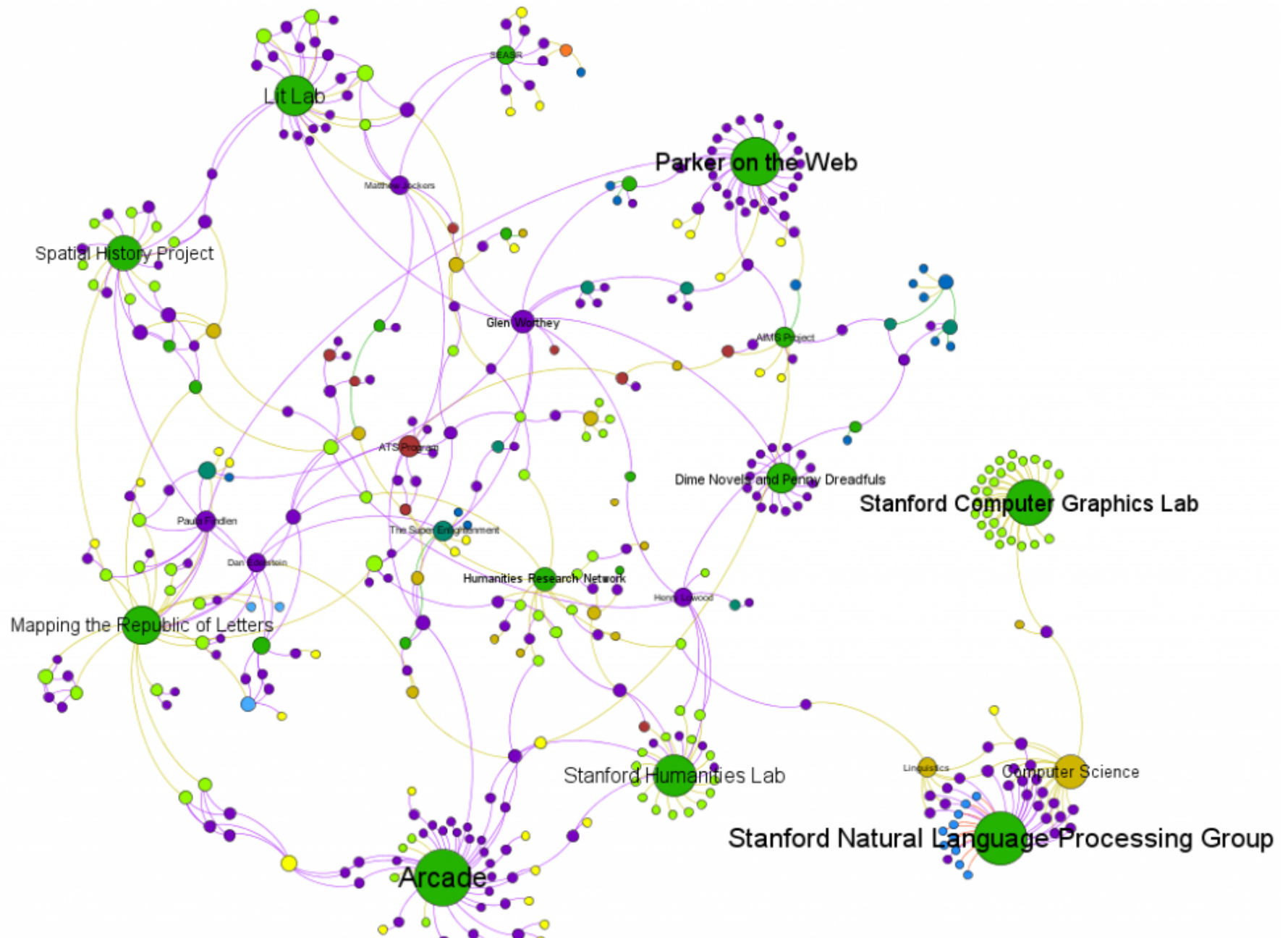
Straight lines

Bending
or curved

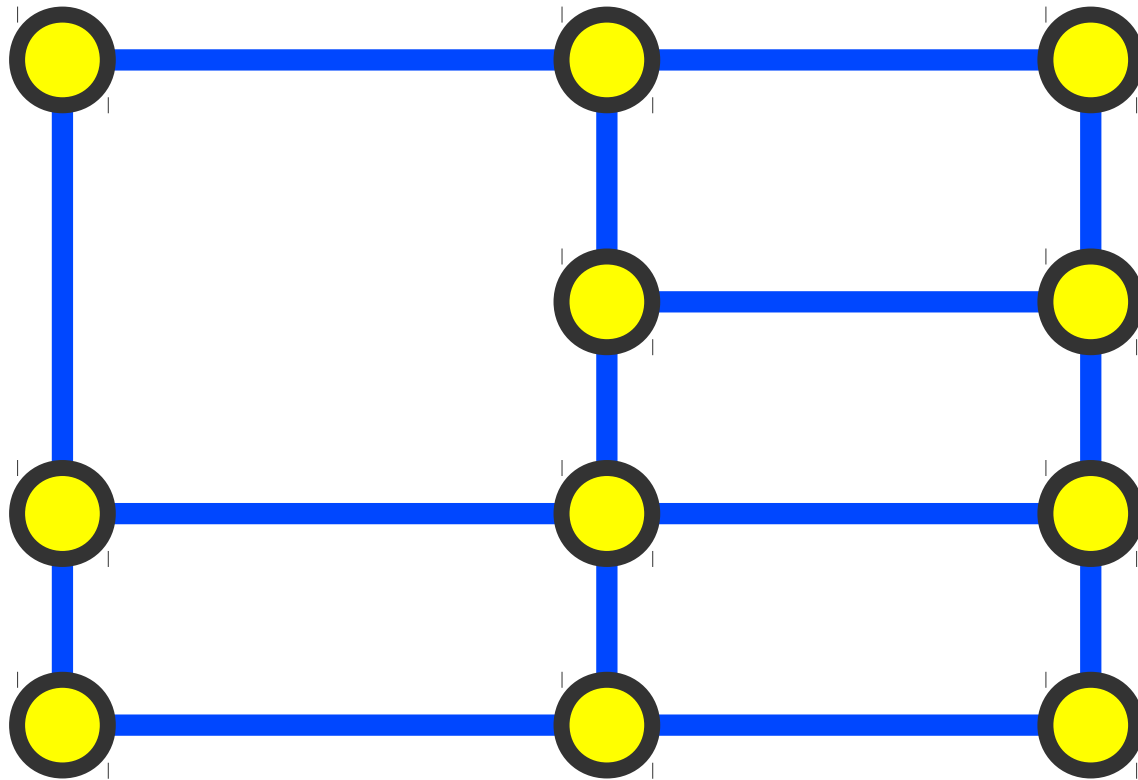
Vertical or
horizontal

Slanting
or diagonal

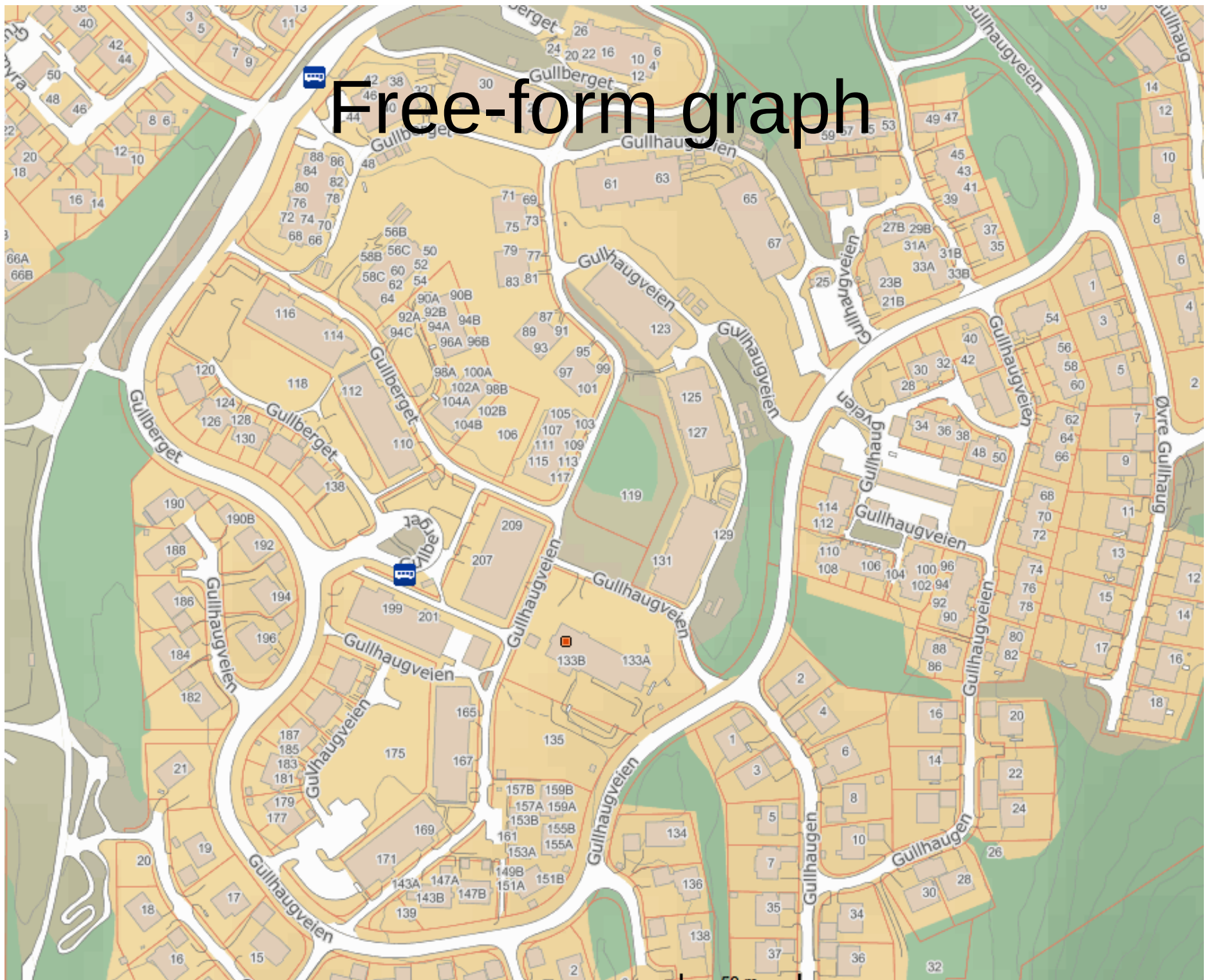
Free-form graph



Rectangular planar graph



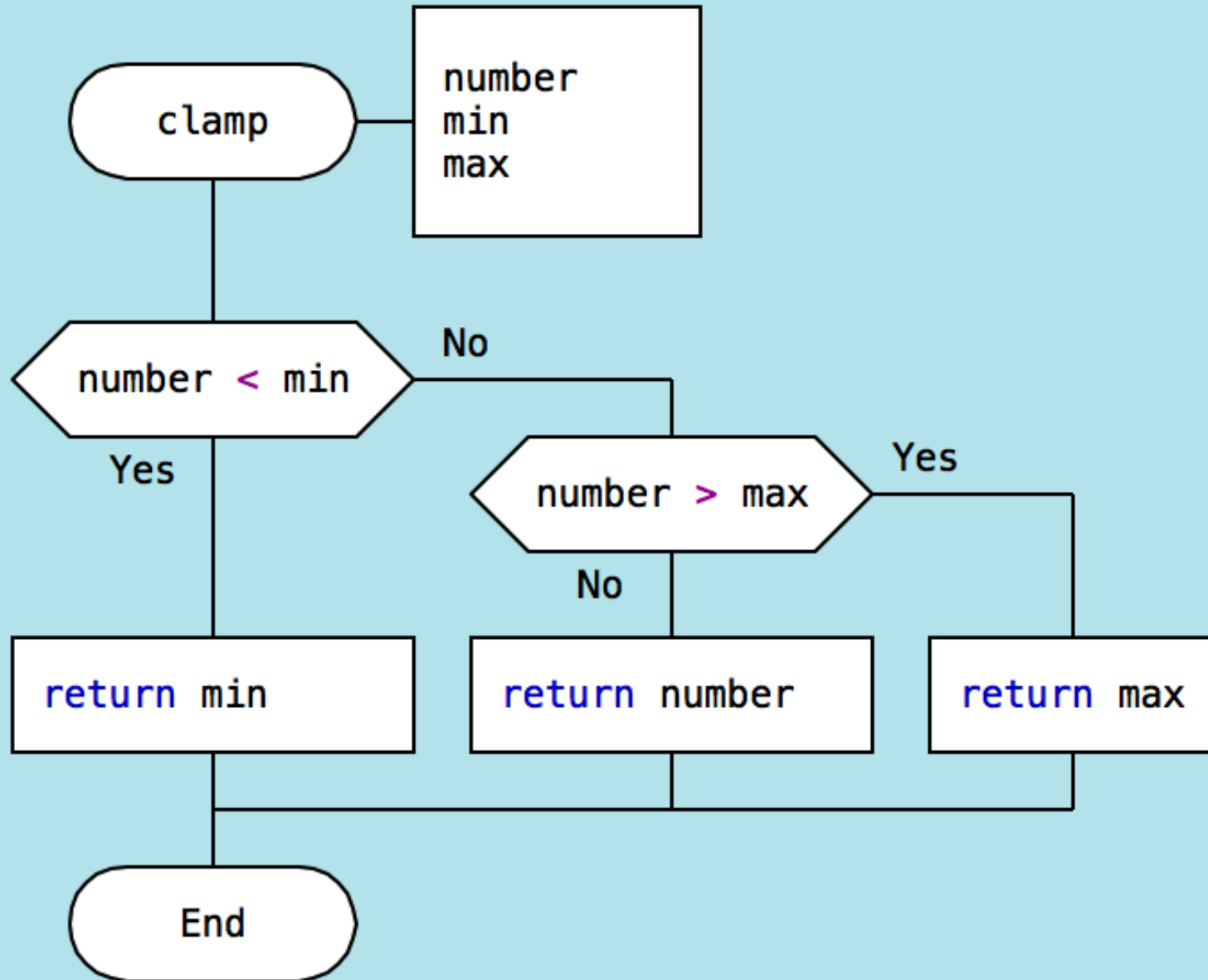
Free-form graph



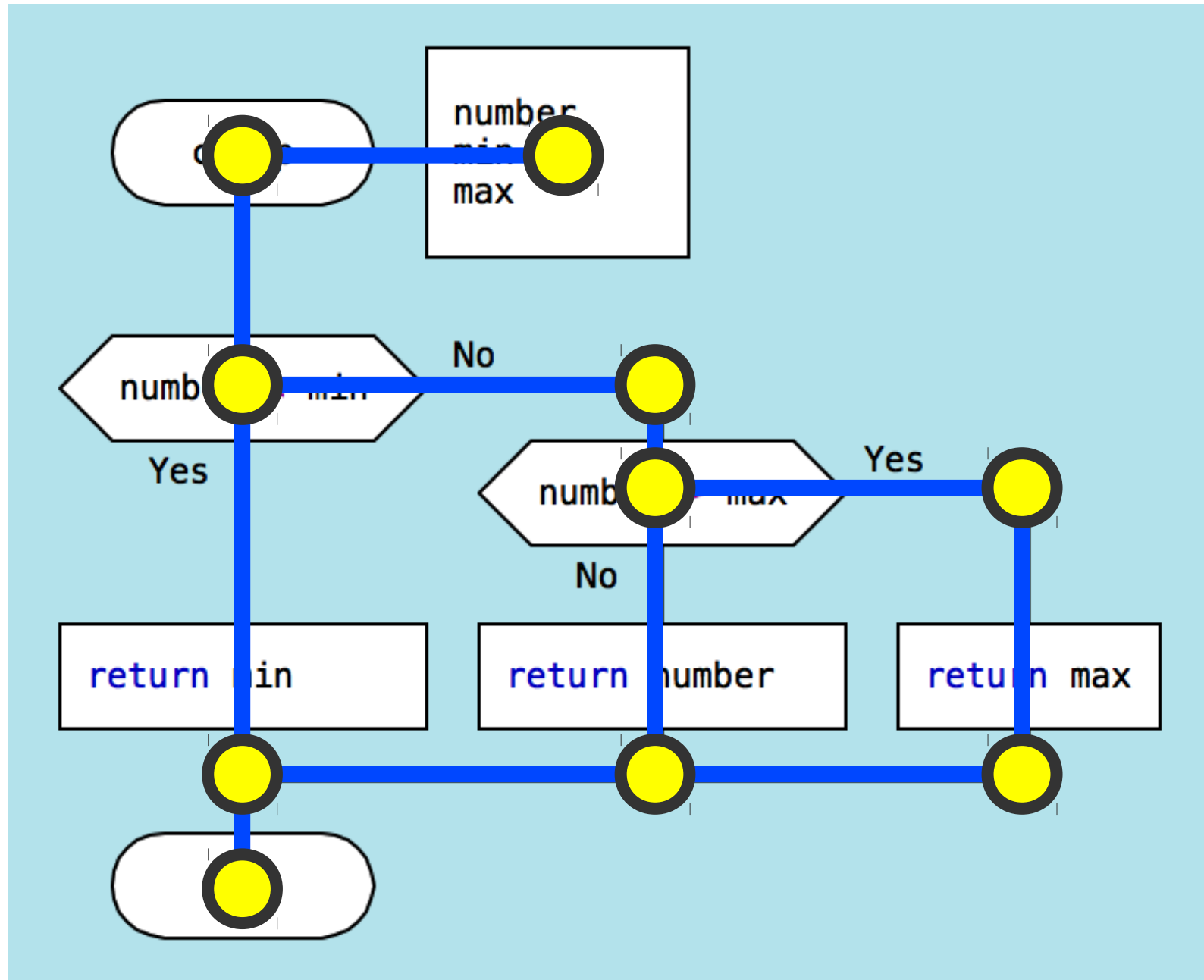
Rectangular planar graph



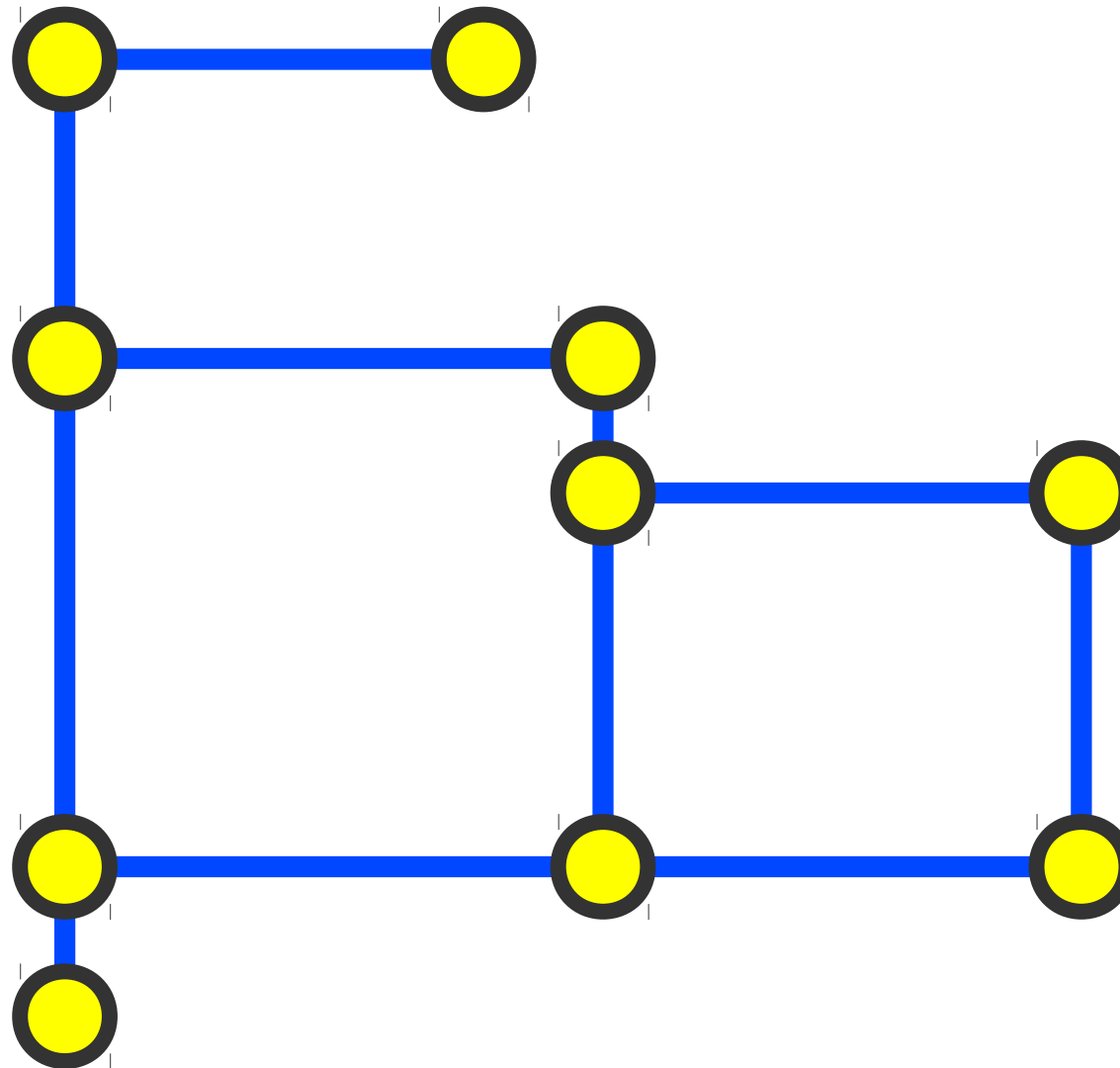
Rectangular planar graph



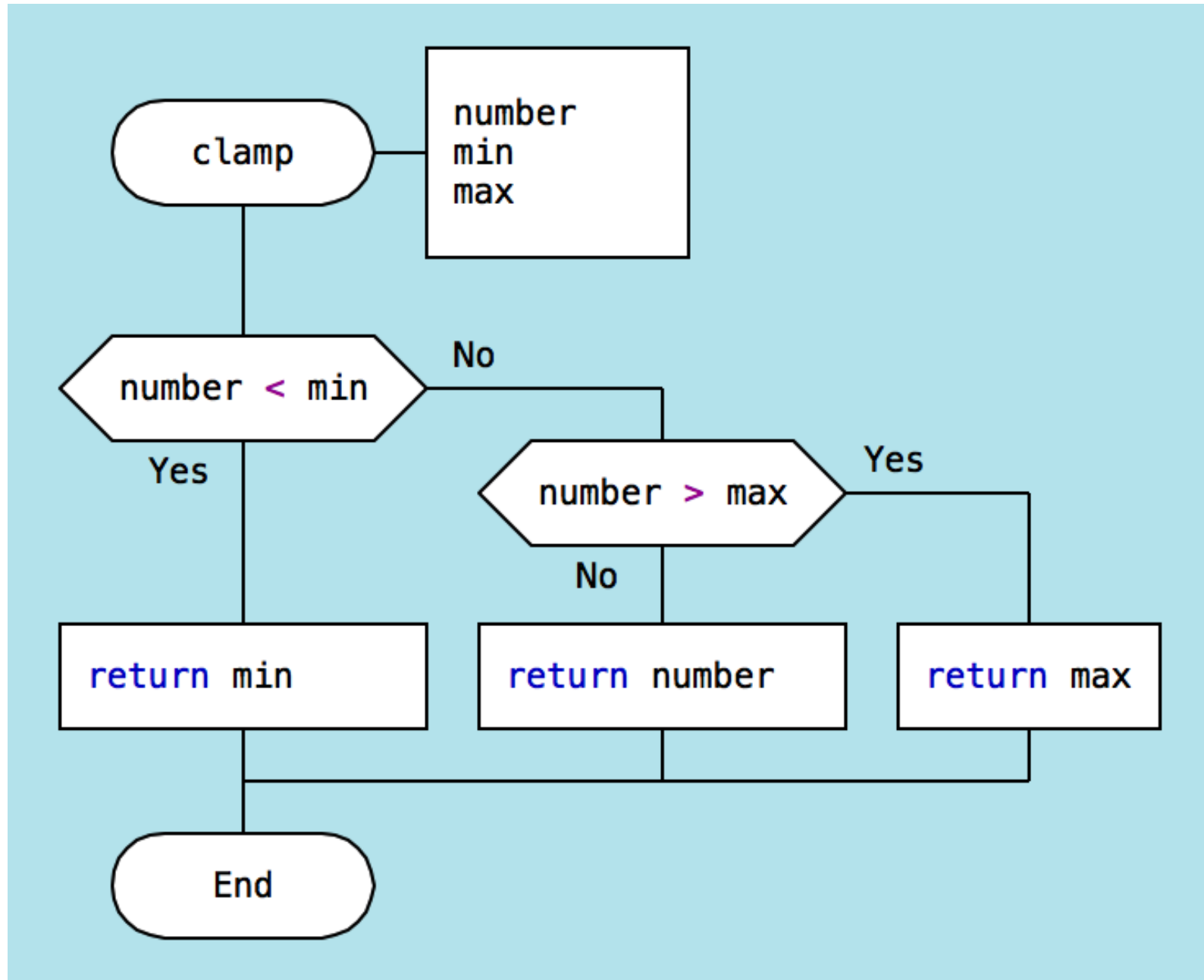
Rectangular planar graph



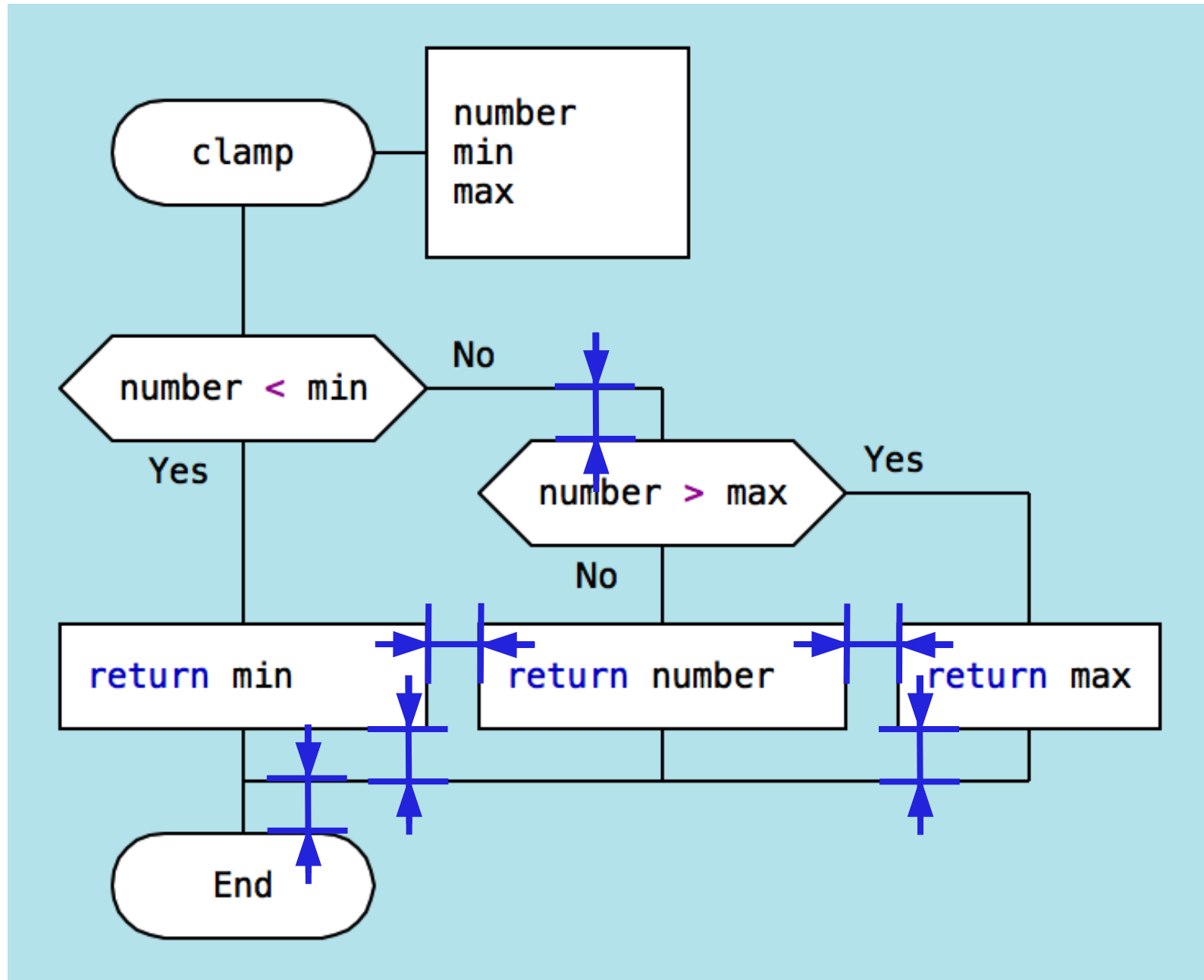
Rectangular planar graph



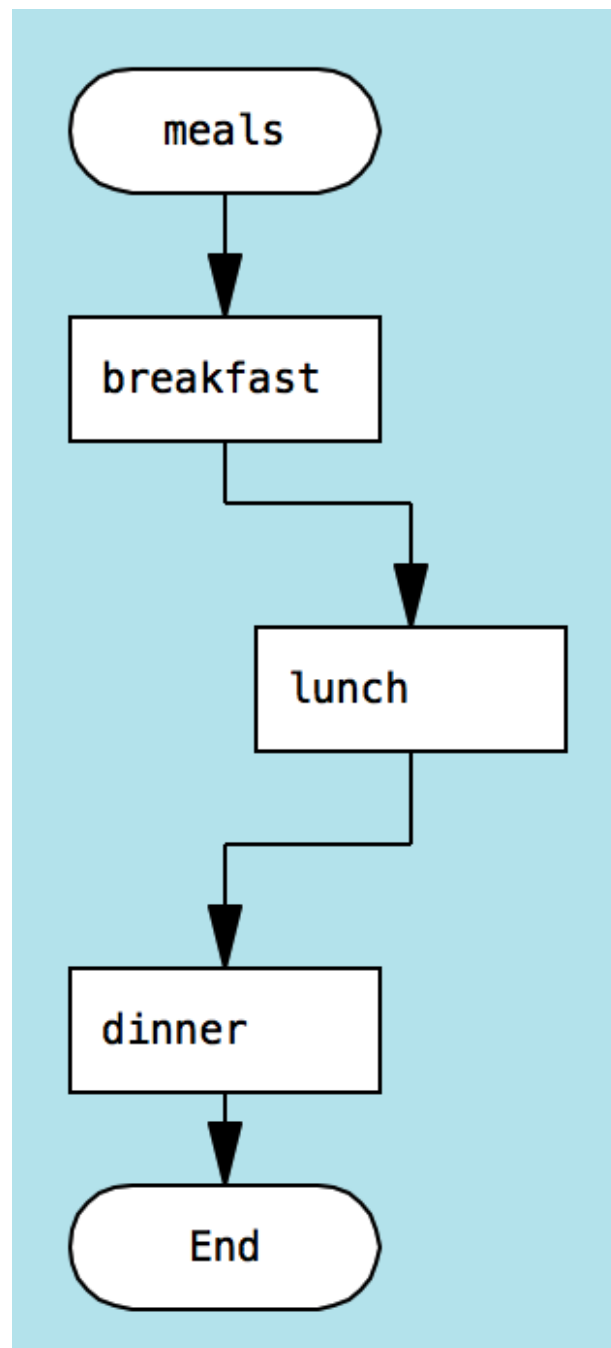
Metre



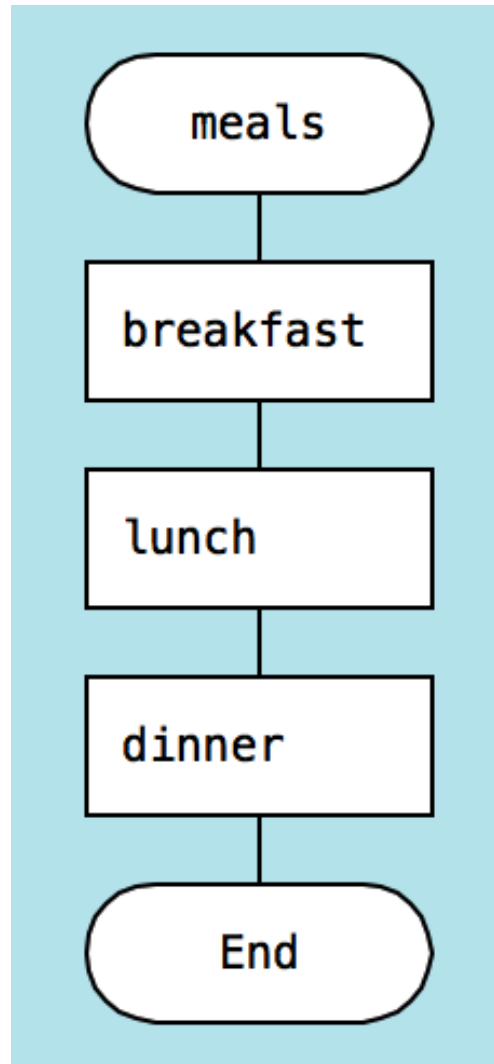
Metre



Visual noise



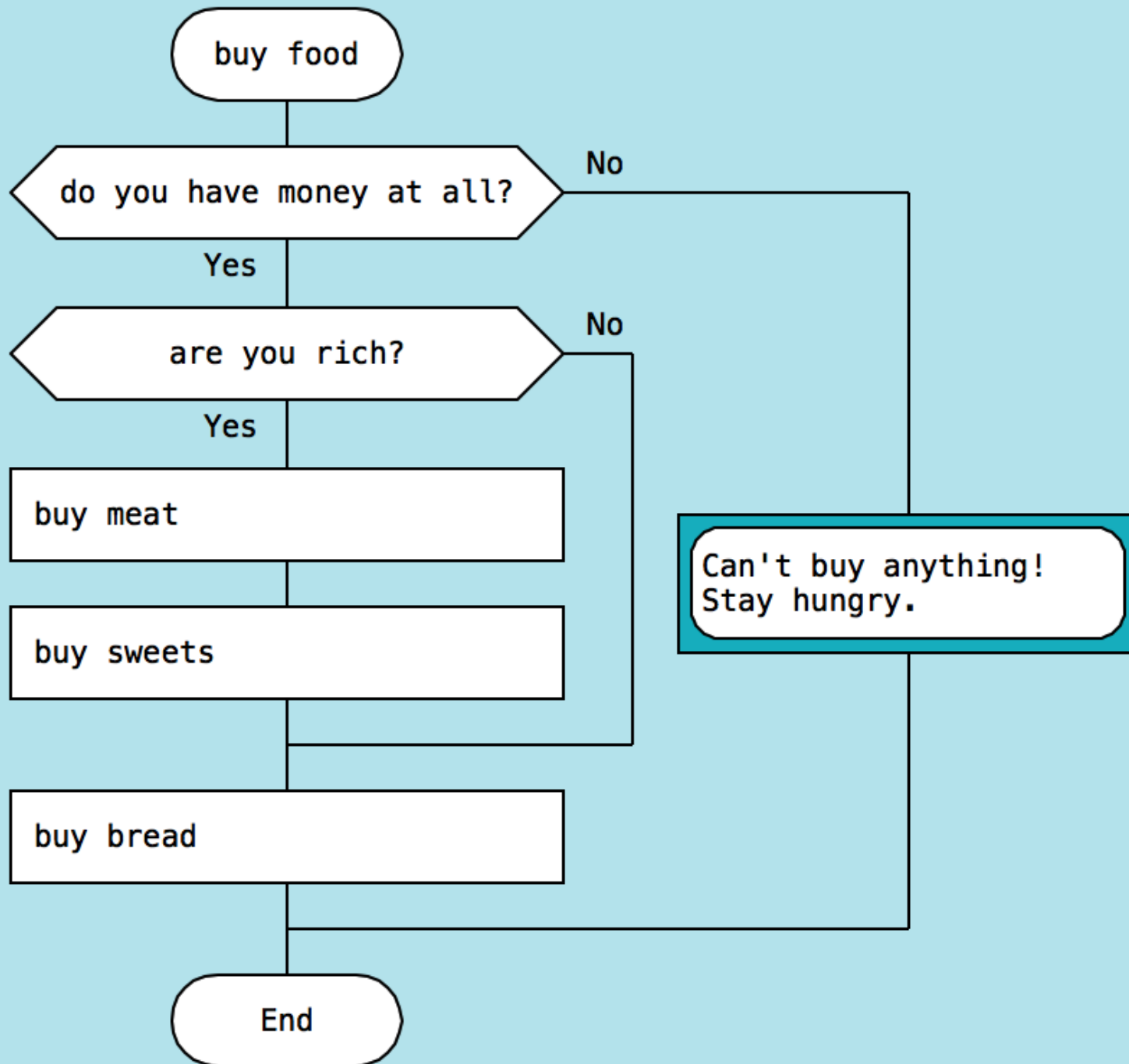
Visual noise removed



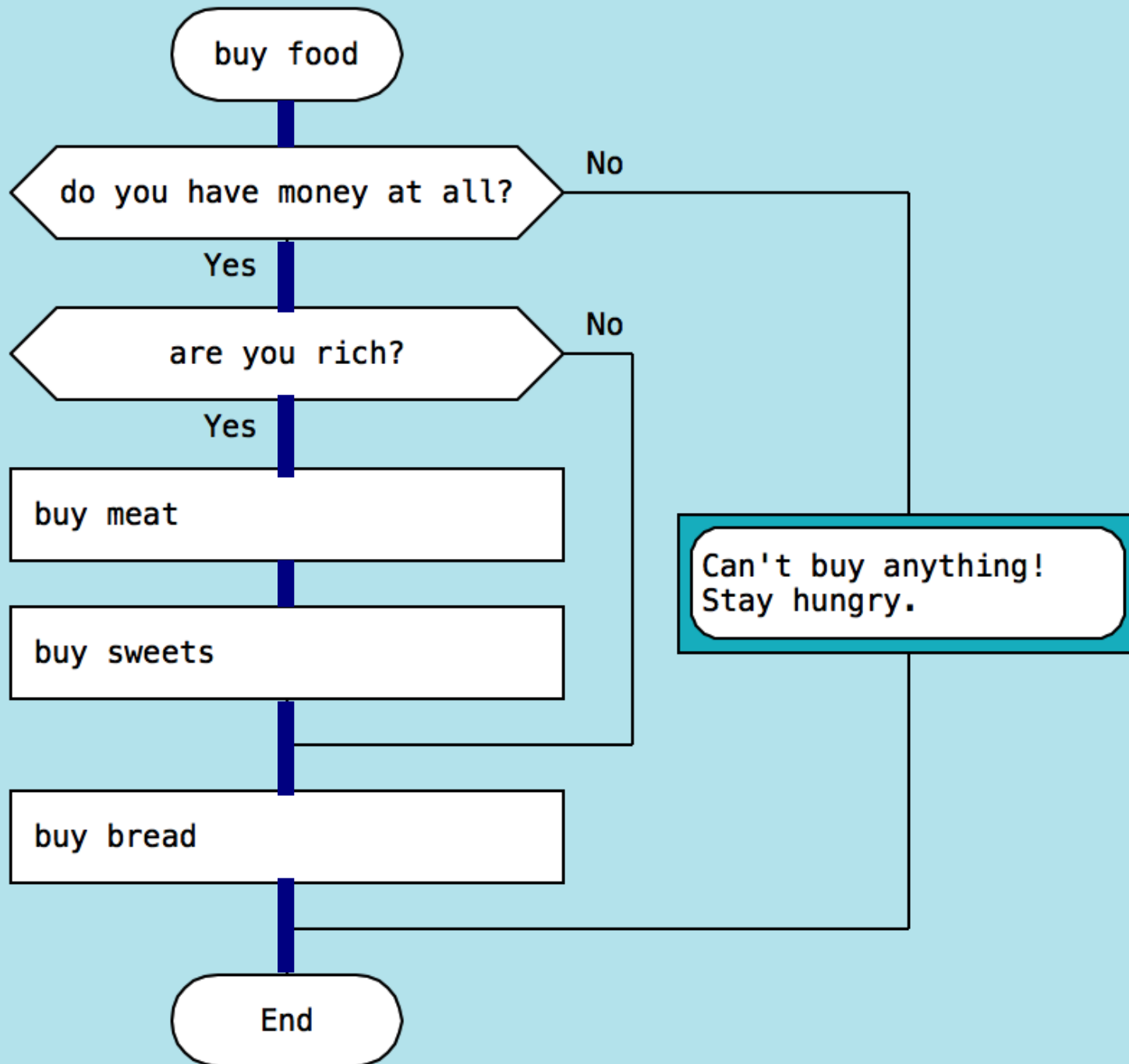
Unique features of DRAKON

- Right is worse
- Silhouette
- Common fate

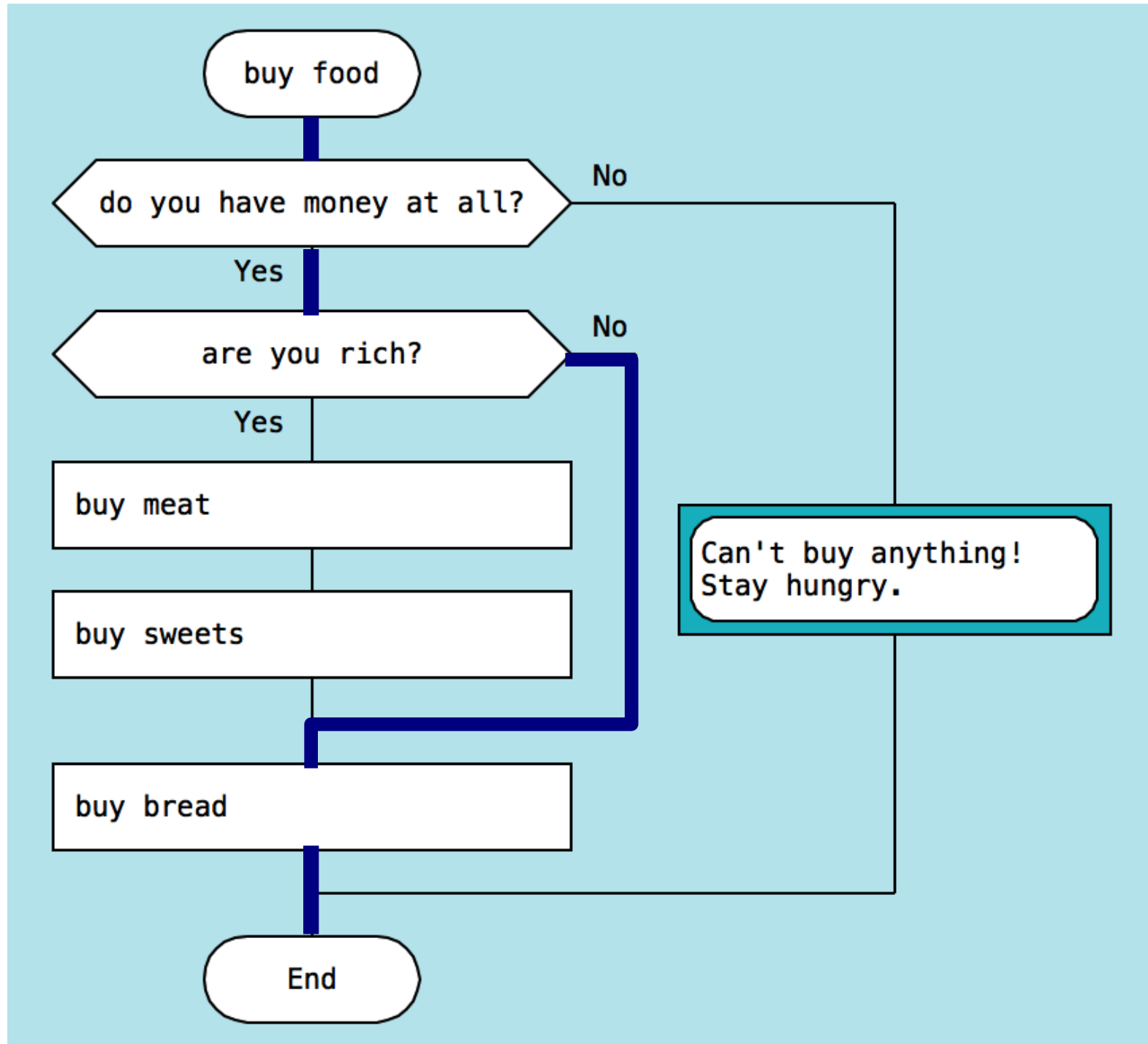
Right is worse



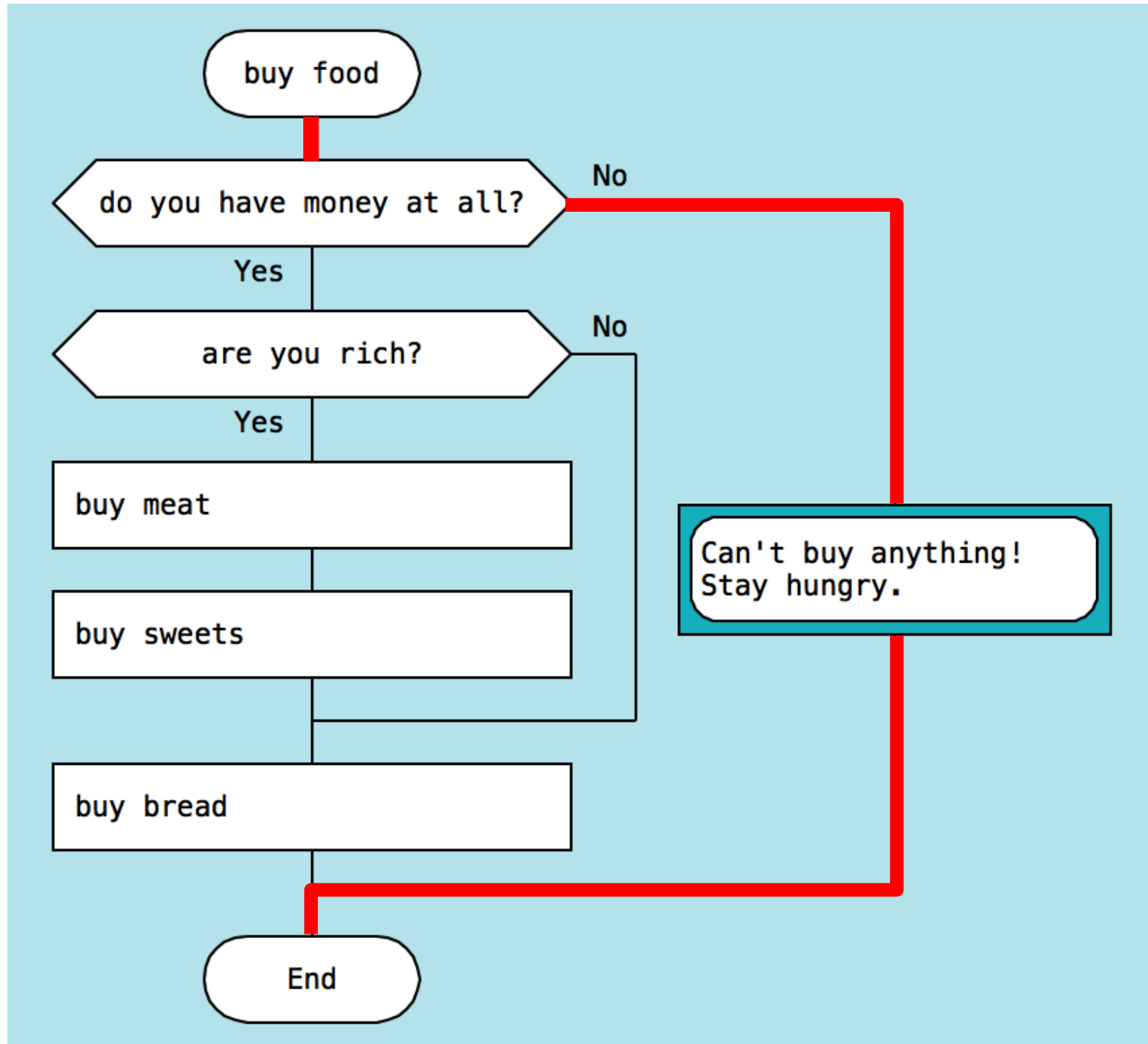
Skewer: the happy path



A worse scenario



Error handling: the worst case

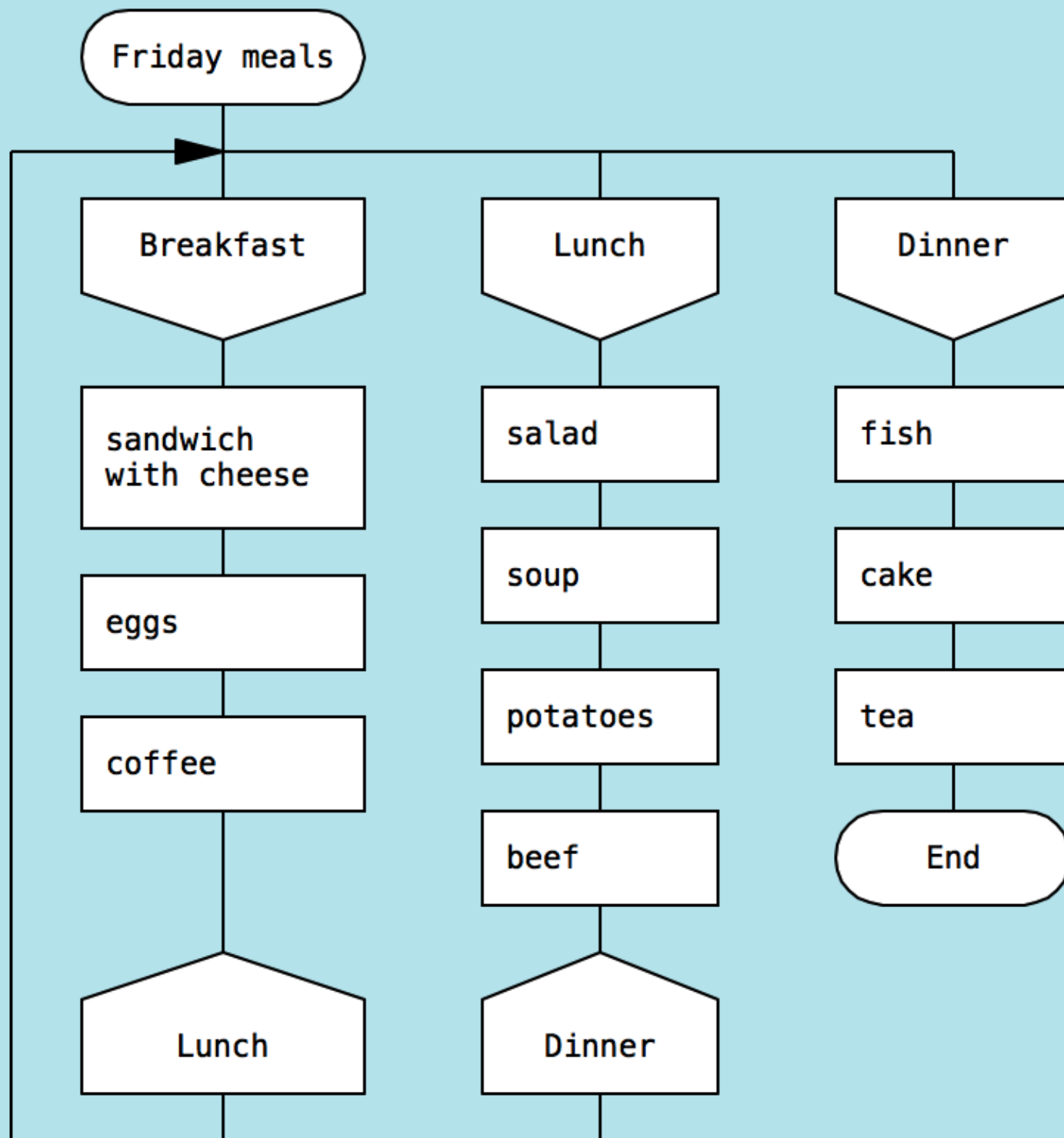


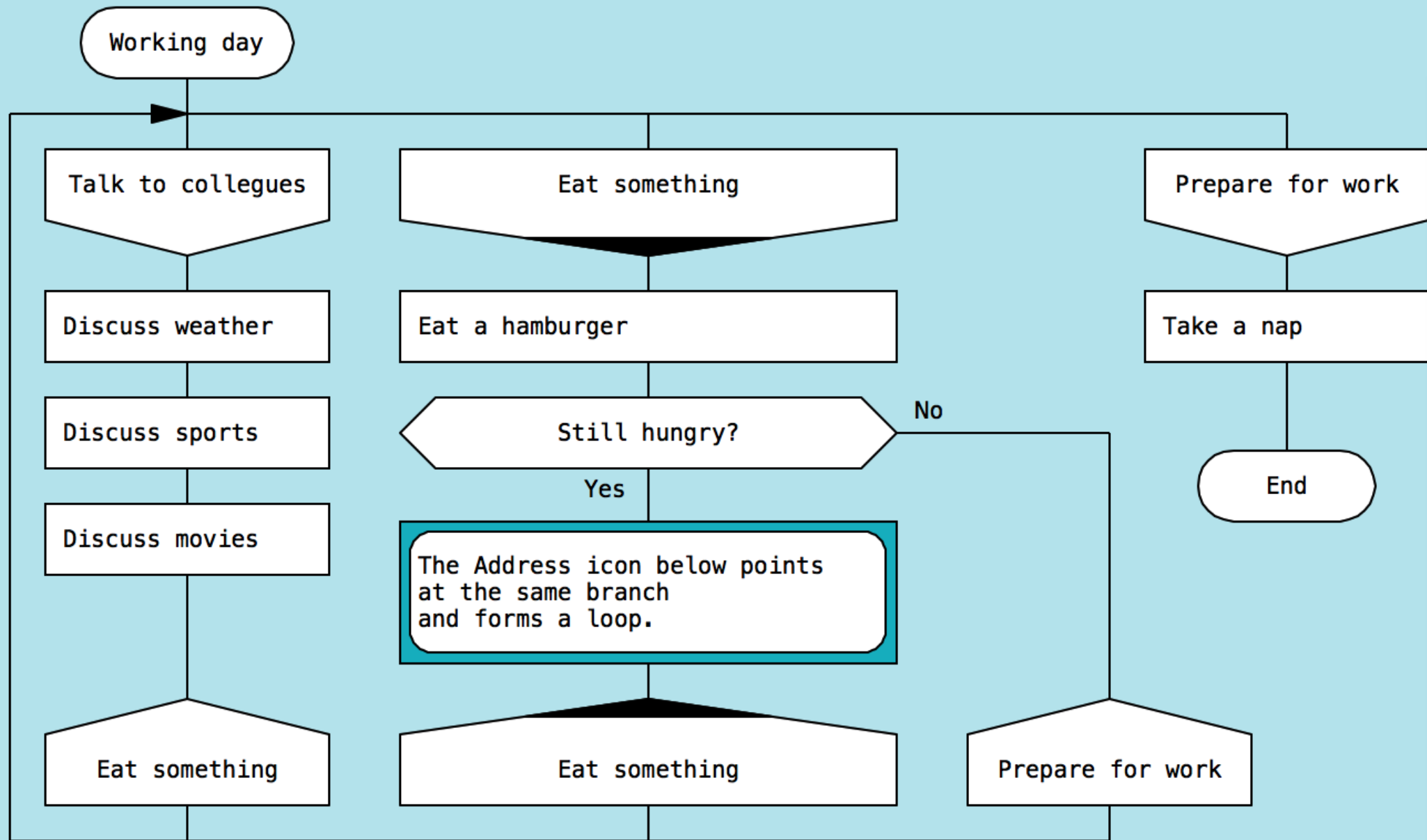
Silhouette

Logical parts

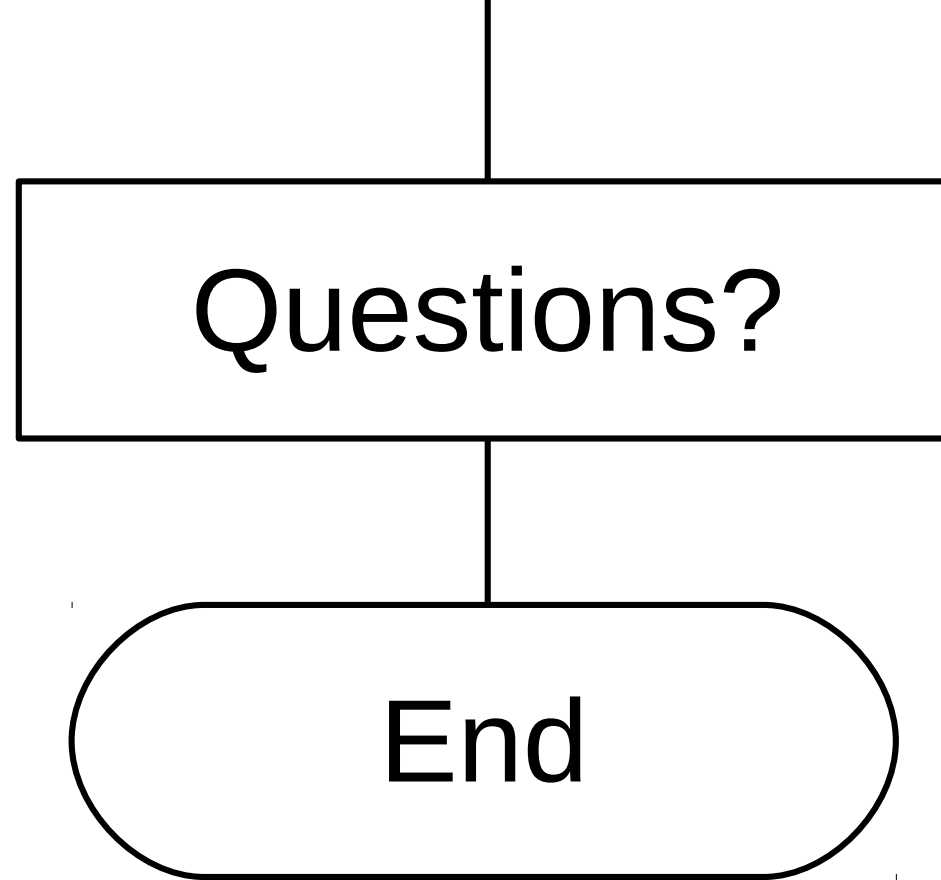
Loops

Finite automata





Seeing the goal brings success



<http://drakon-editor.sourceforge.net/>

http://www.slideshare.net/stepan_mitkin/drakon-part1-eng