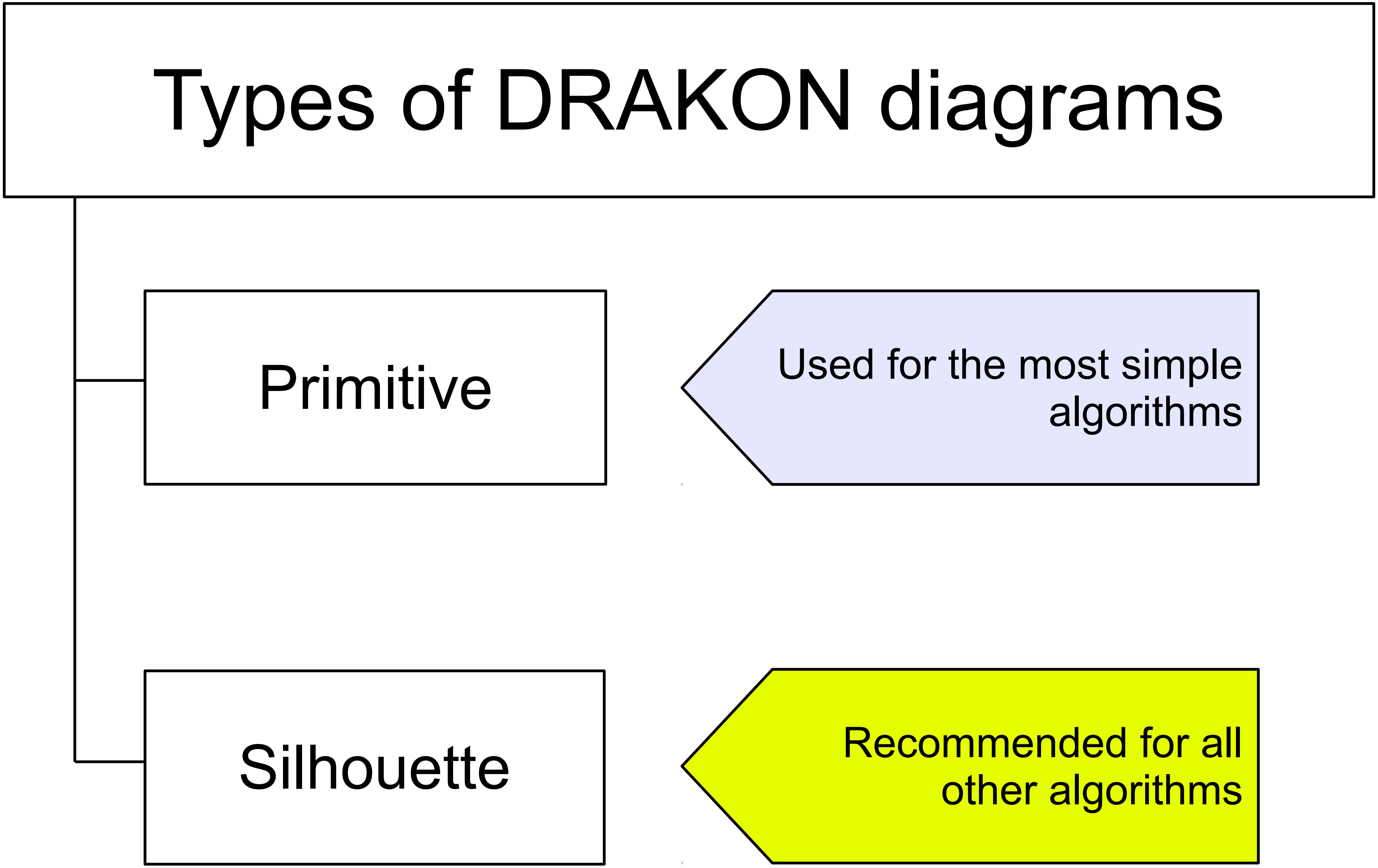
DRAKON Visual Language

Tutorial. Part 1:
How to draw DRAKON flowcharts



Types of DRAKON diagrams

Primitive

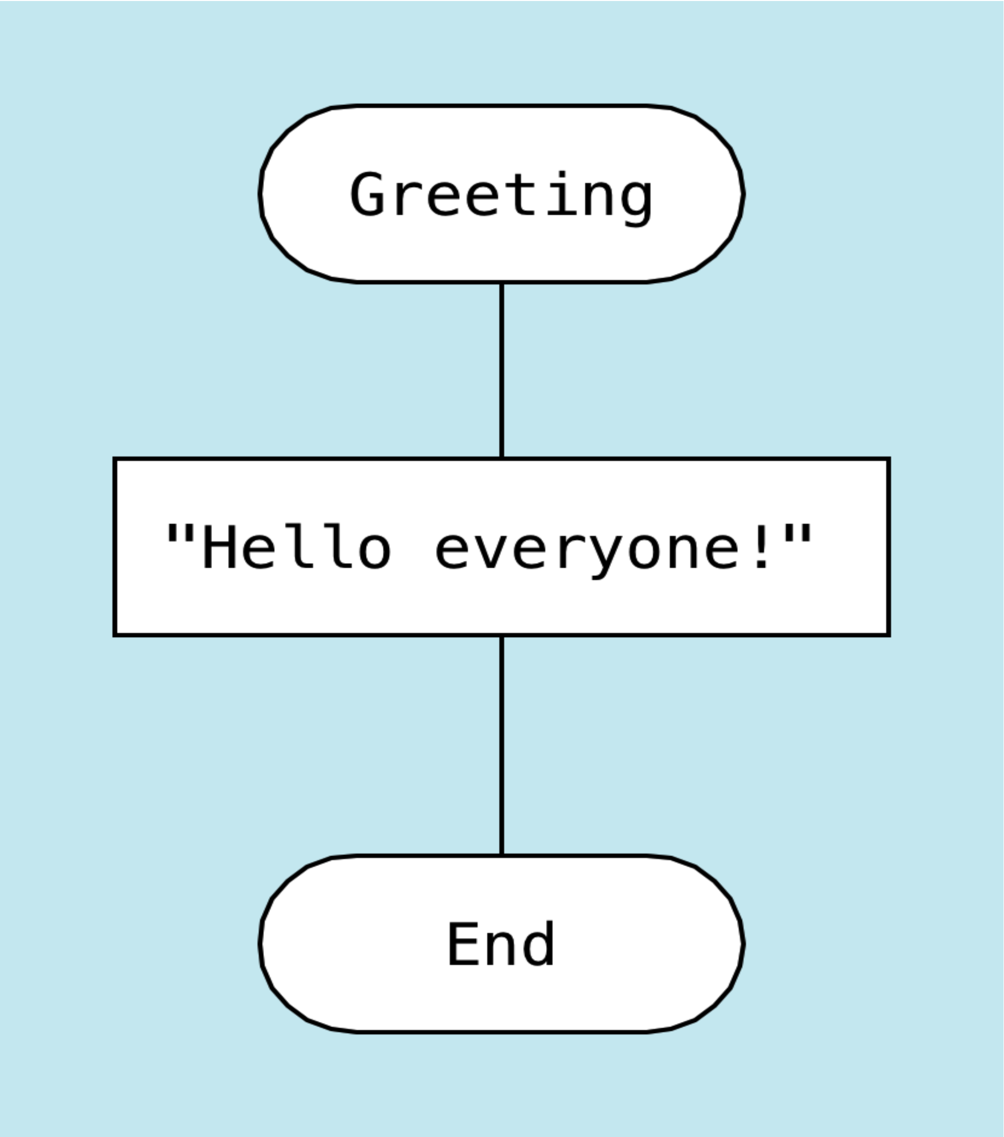
Used for the most simple algorithms

Silhouette

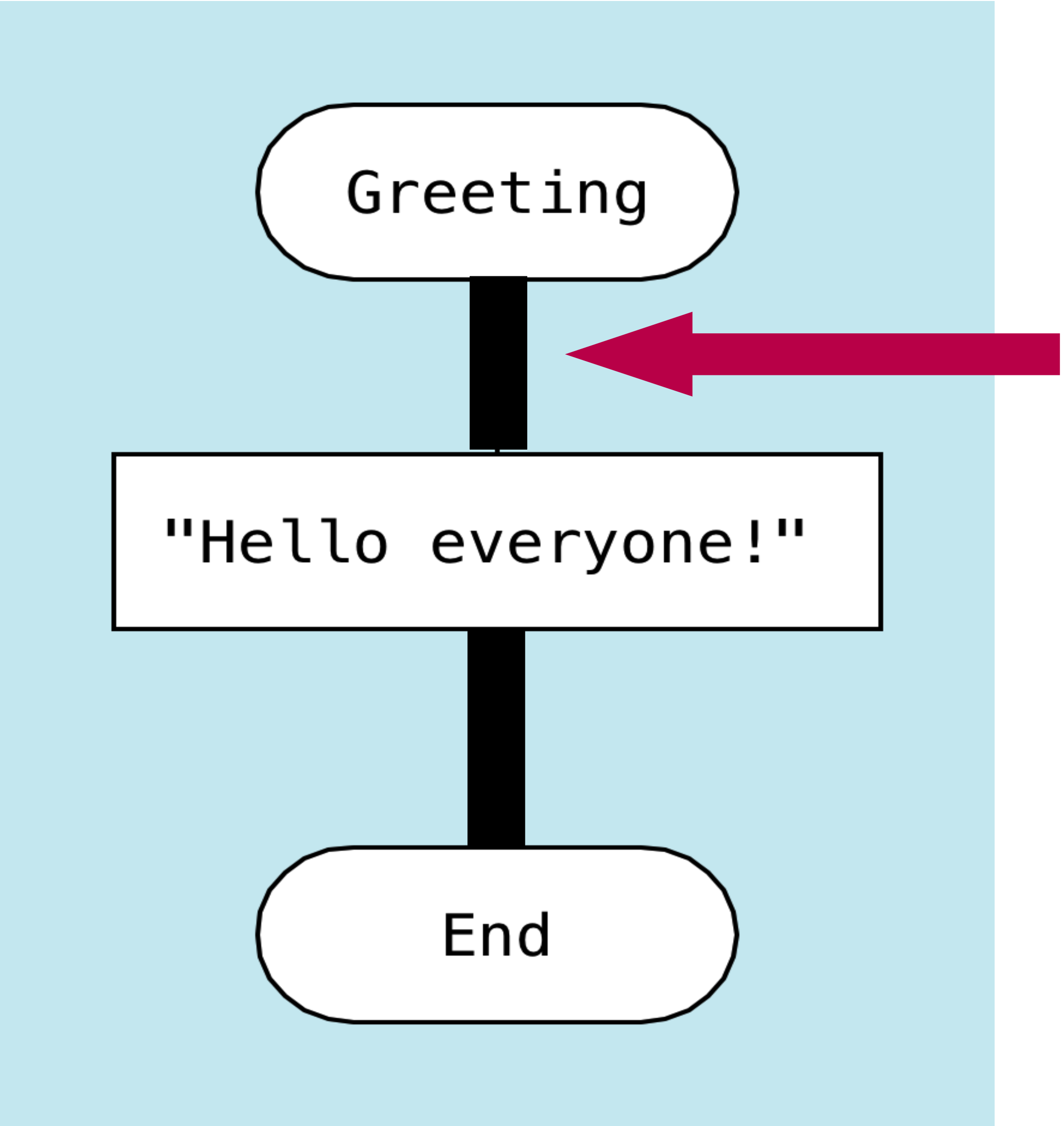
Recommended for all other algorithms

Primitive

What does a primitive consist of?

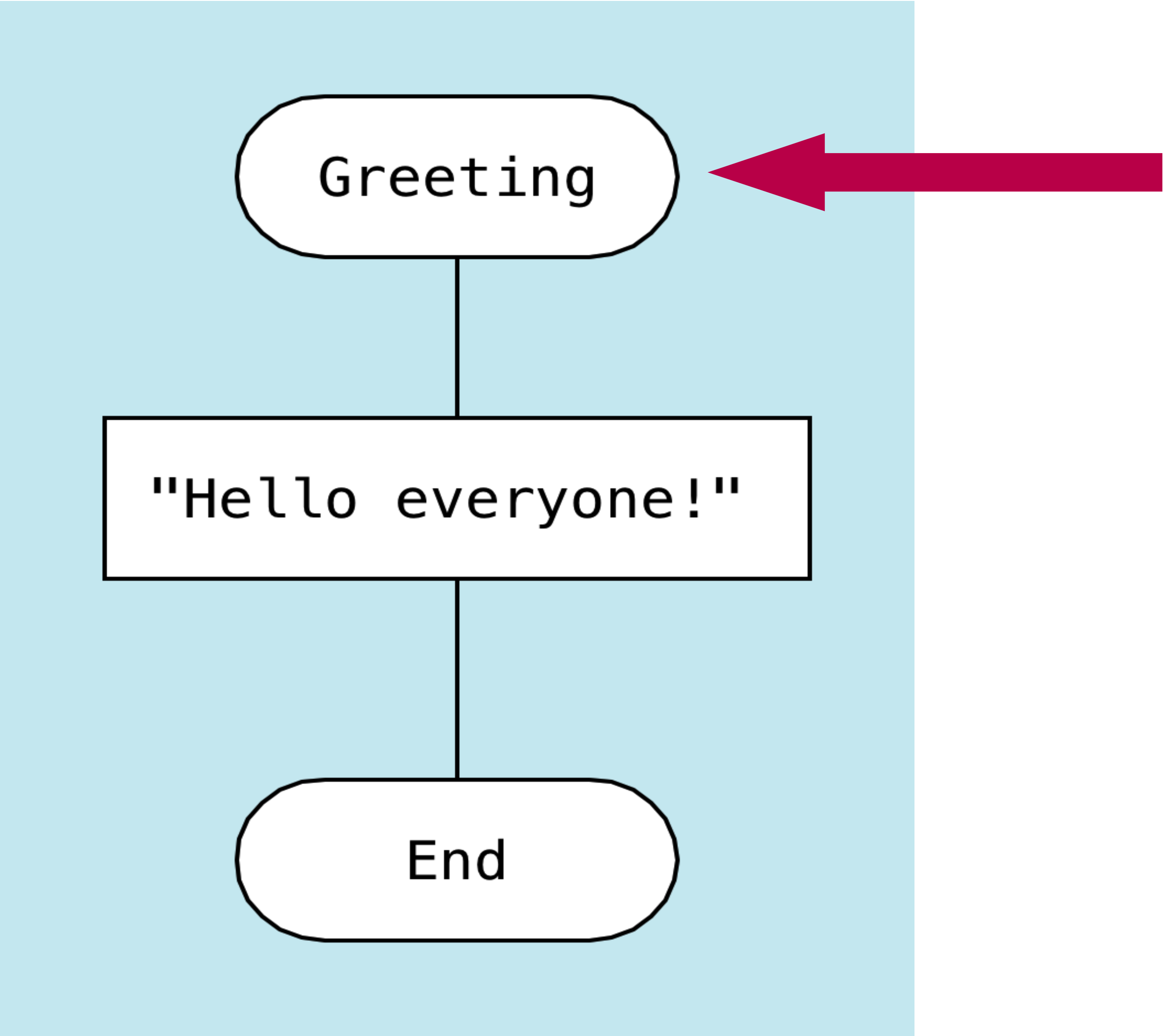


What does a primitive consist of?



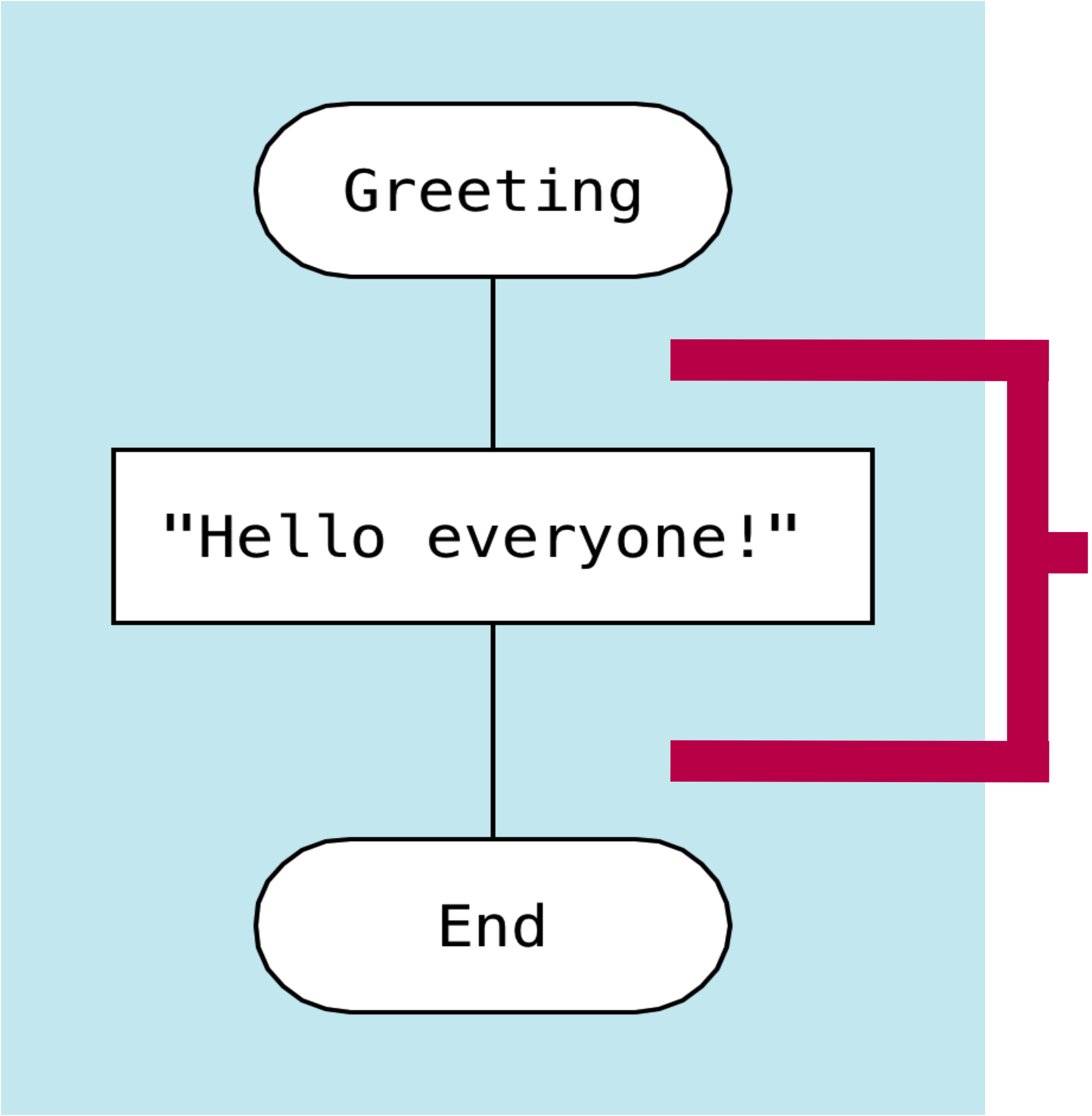
A vertical skewer

What does a primitive consist of?



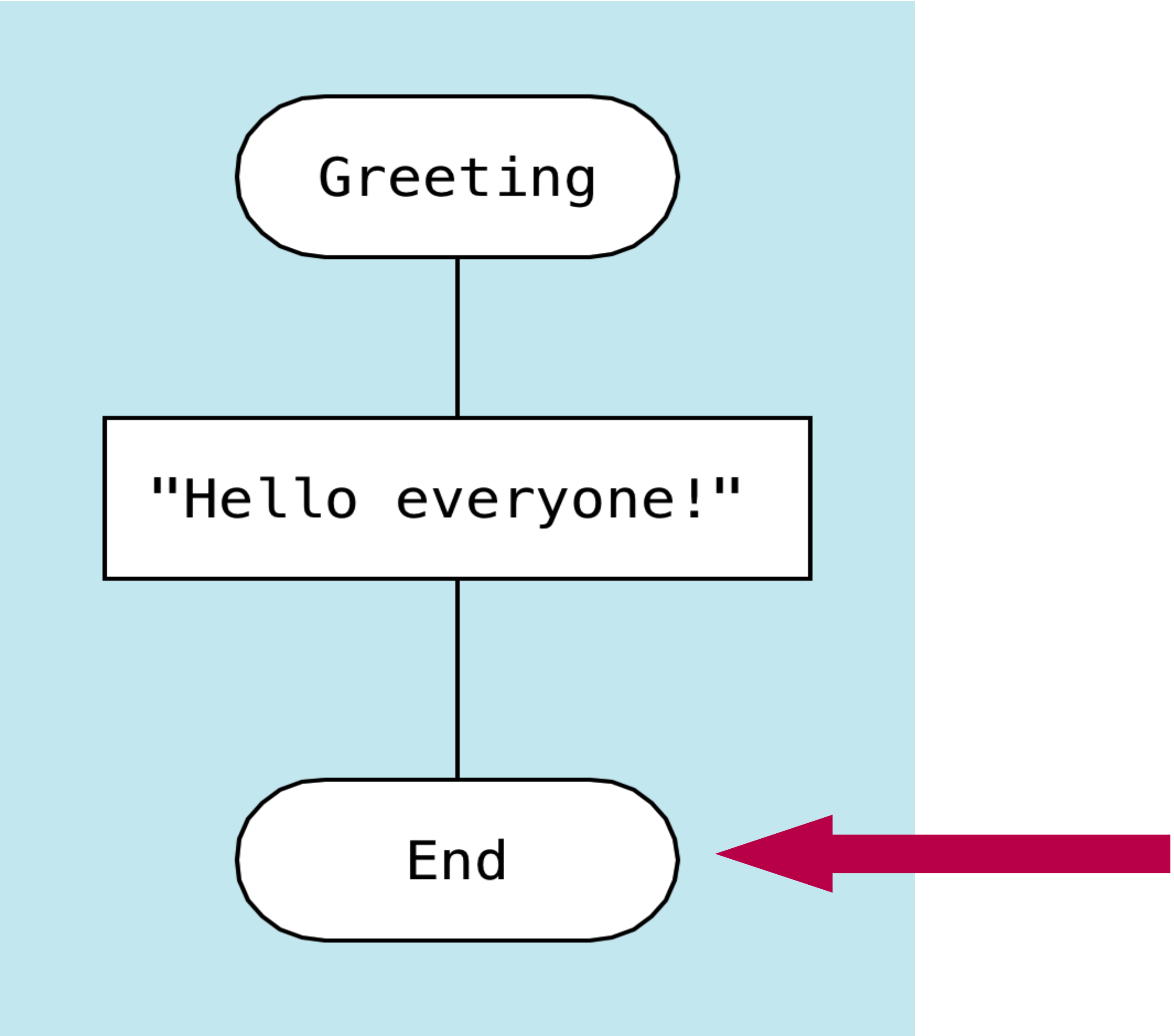
The header

What does a primitive consist of?



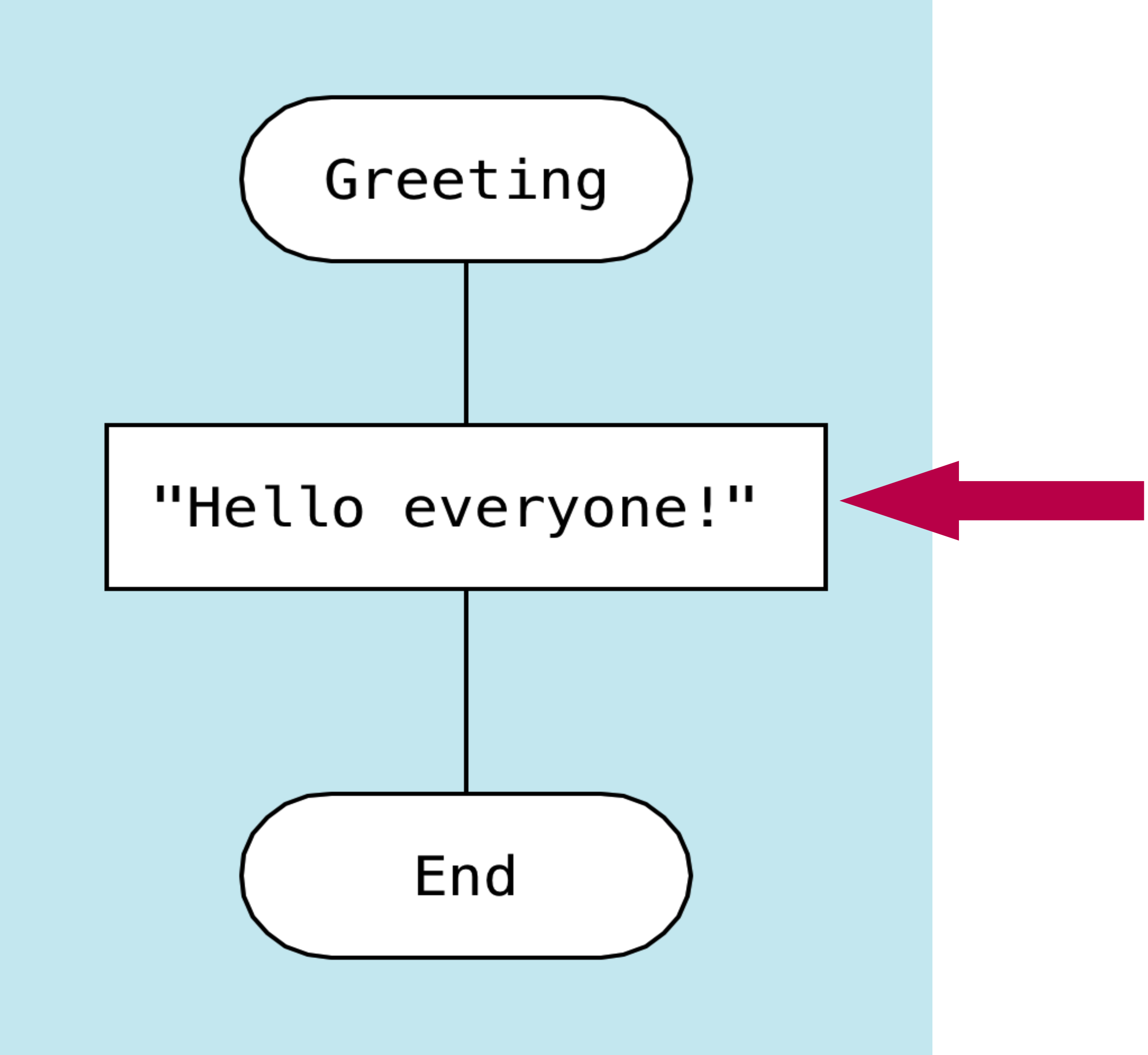
The algorithm

What does a primitive consist of?



The “End” icon

“Action” icon

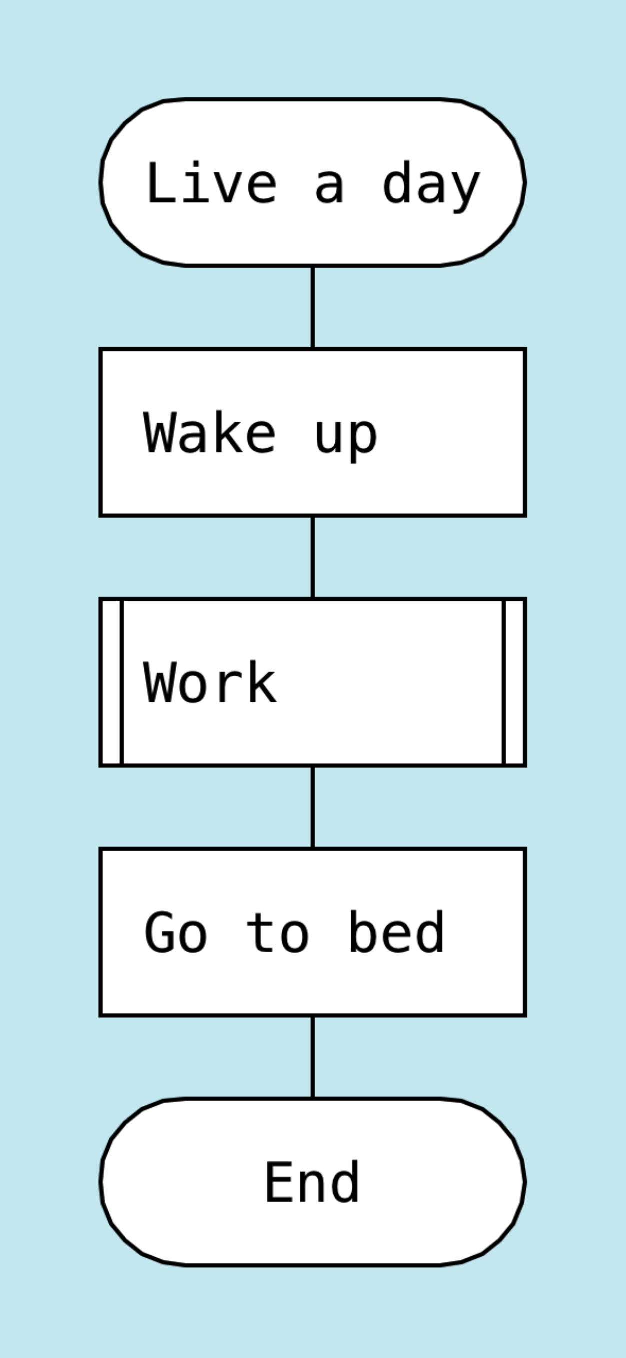


An “Action” icon

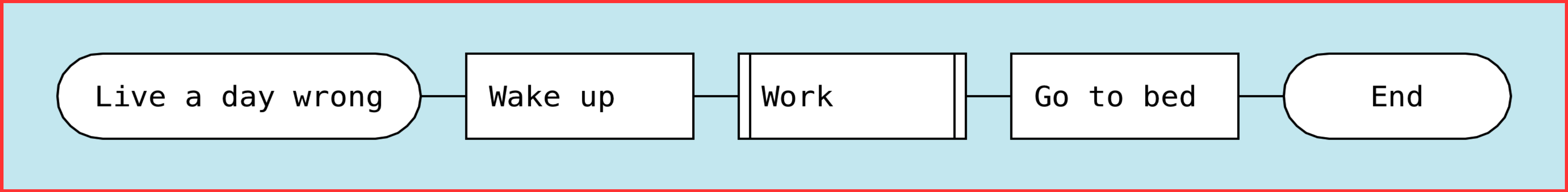
“Action” icon

an order to do something

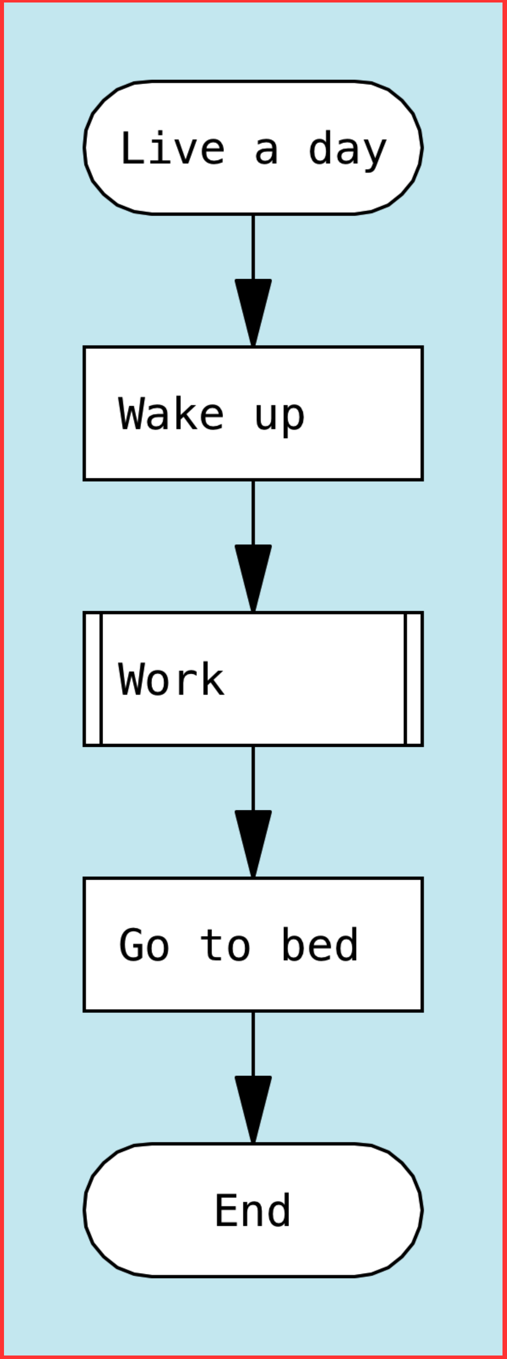
The skewer must be vertical



Wrong!



Wrong again

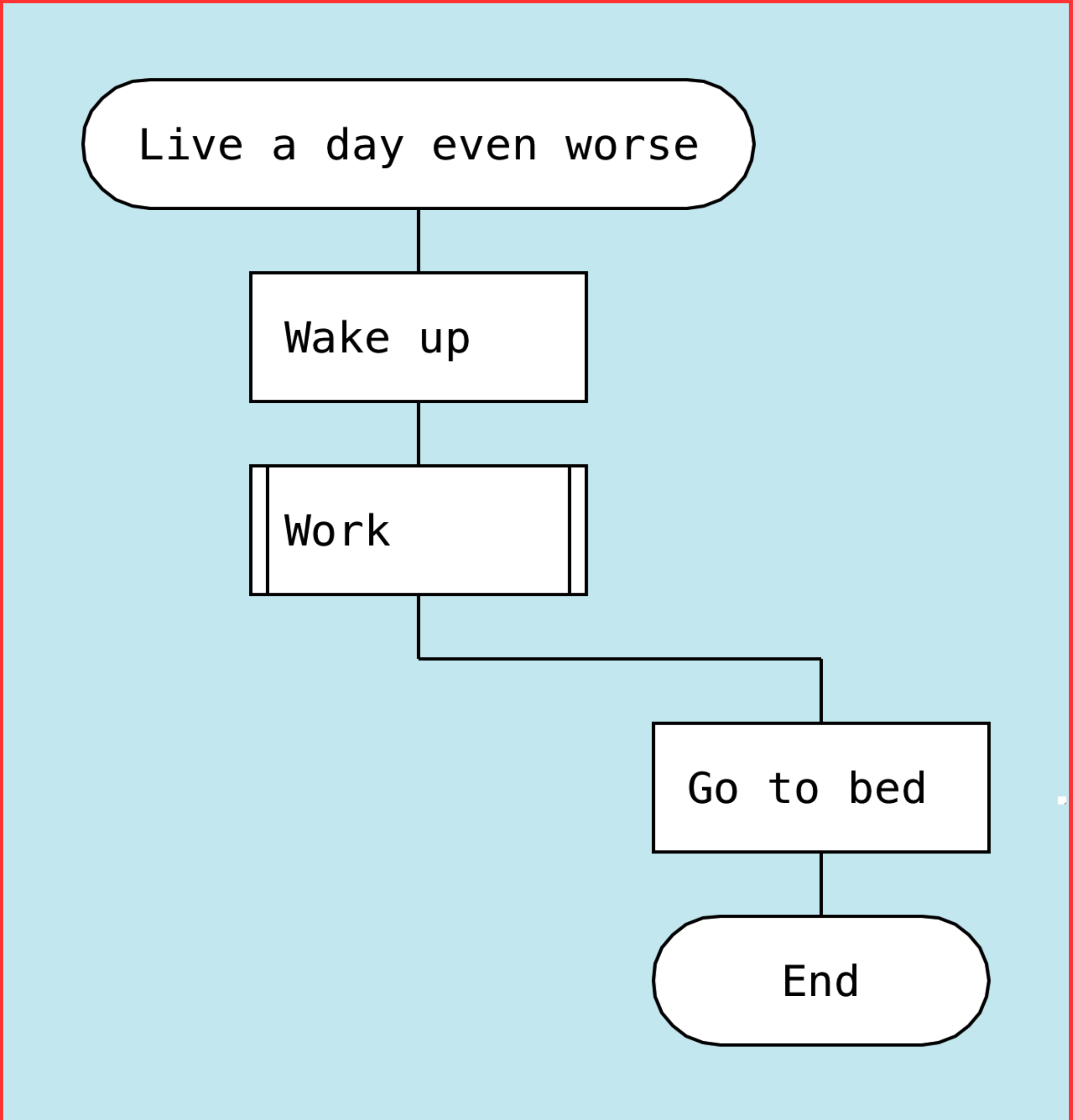


- Use lines instead of arrows
- Arrows create visual noise

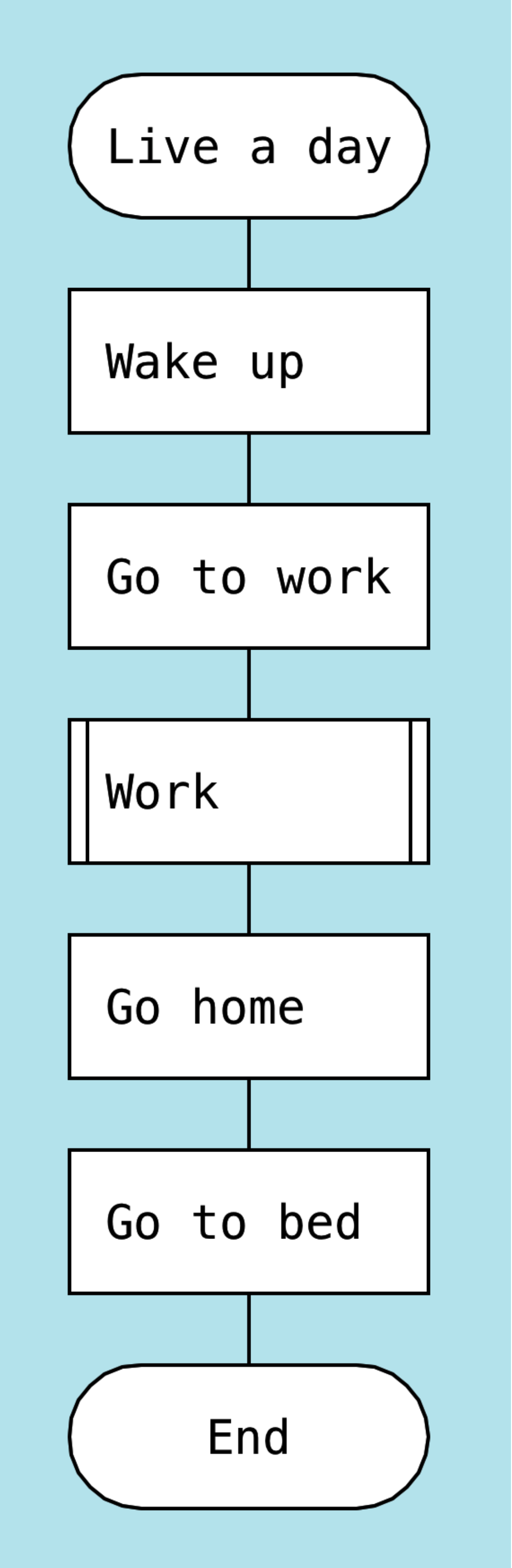
In DRAKON,
the next icon is always below

Even worse

- It is not allowed to break the skewer



“Insertion” icon



Live a day

Wake up

Go to work

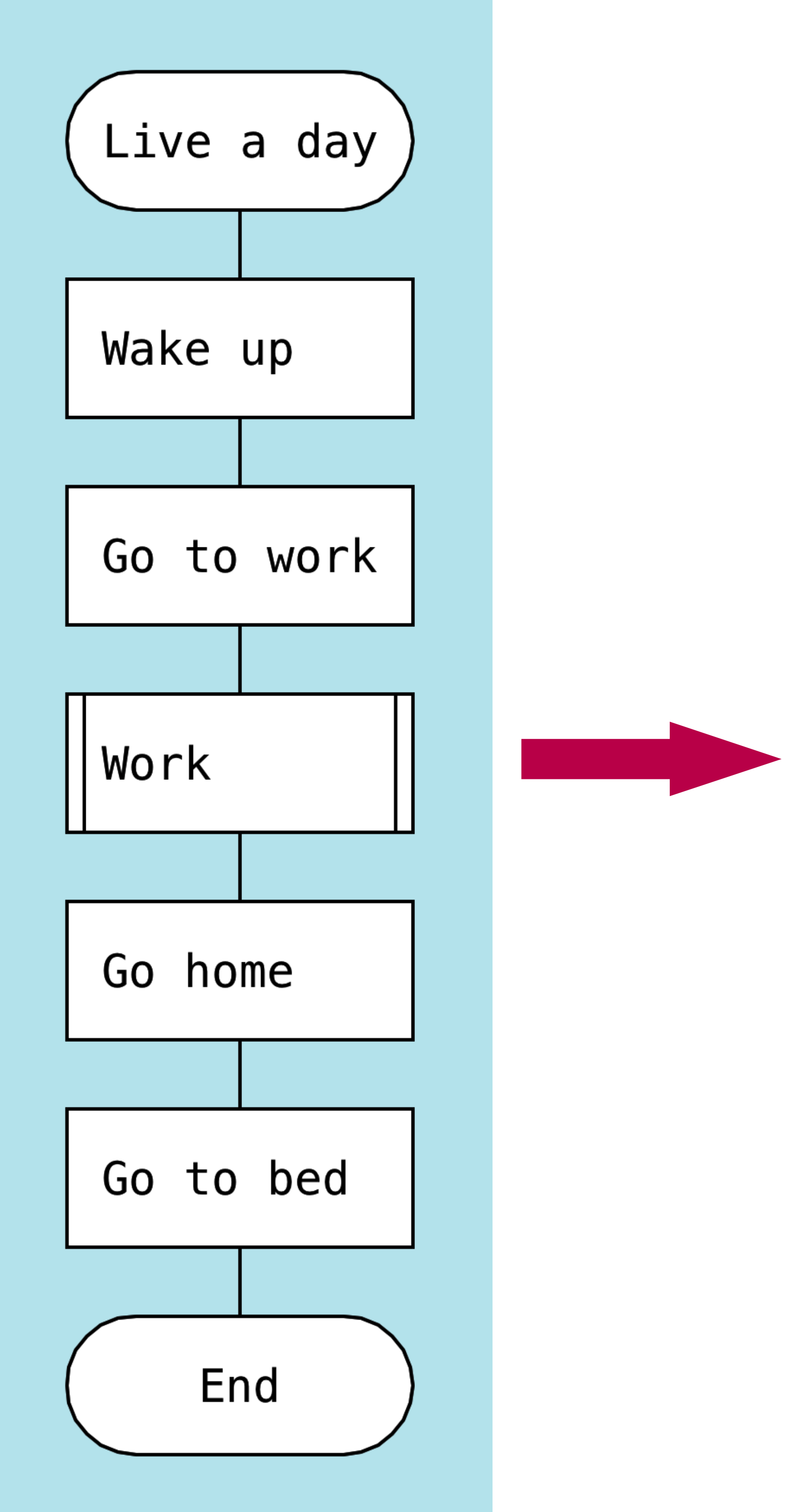
Work

Go home

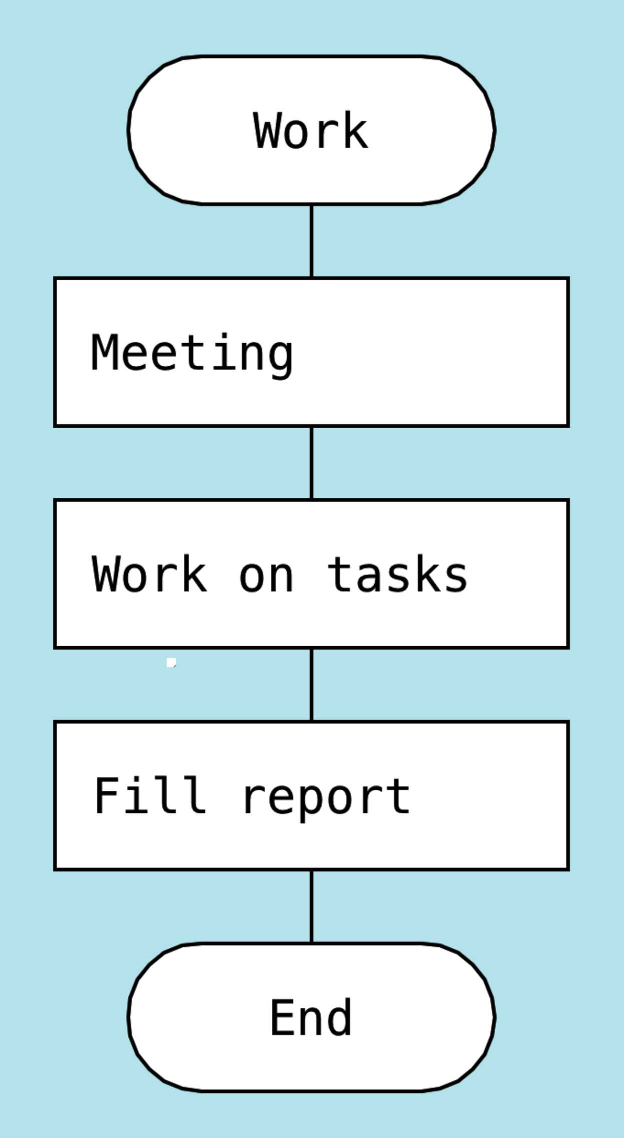
Go to bed

End

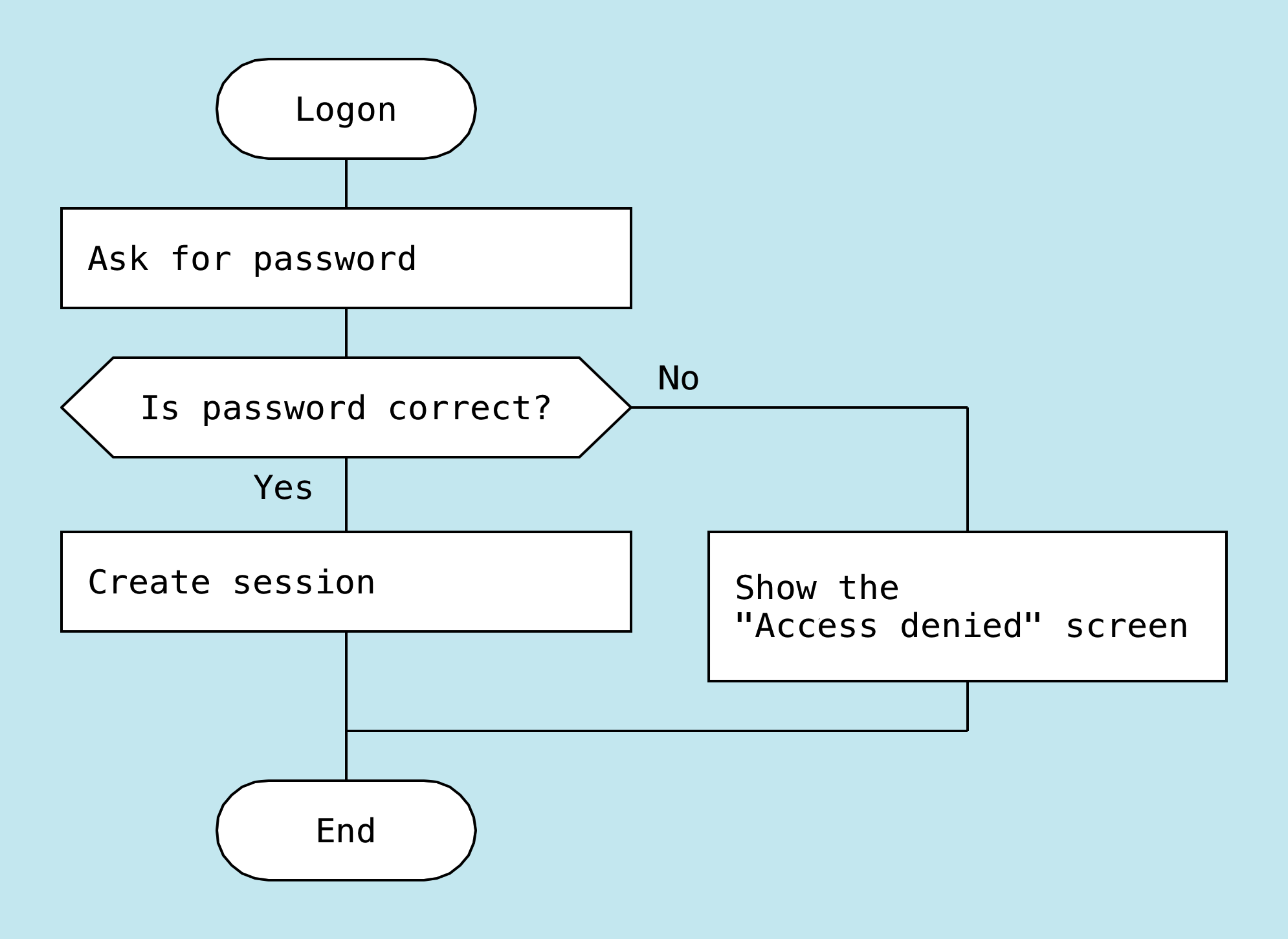
It is a reference
to an algorithm
that is defined elsewhere



“Insertion” icon



“If” icon



Logon

Ask for password

Is password correct?

No

Yes

Create session

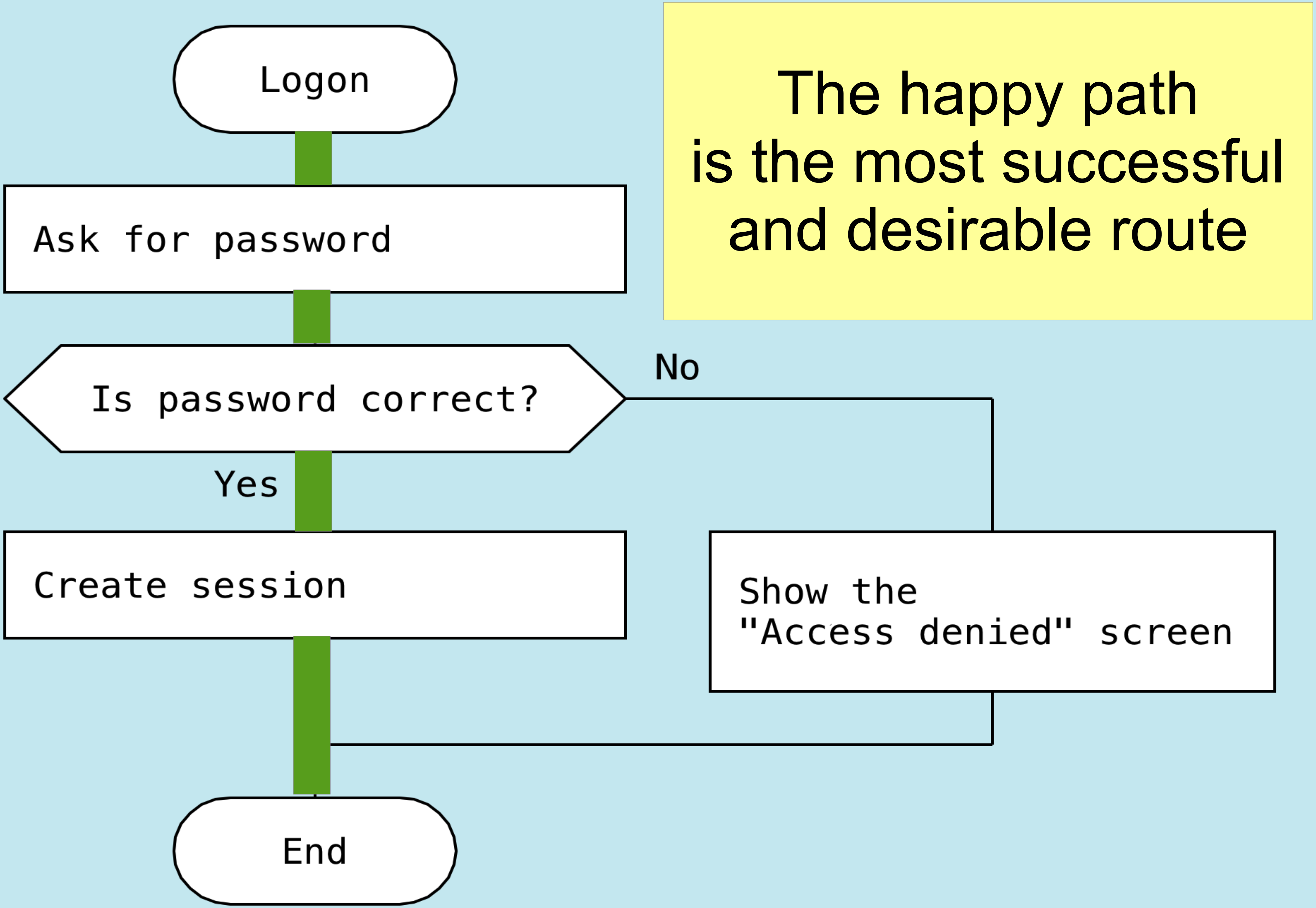
Show the "Access denied" screen

End

“If” icon

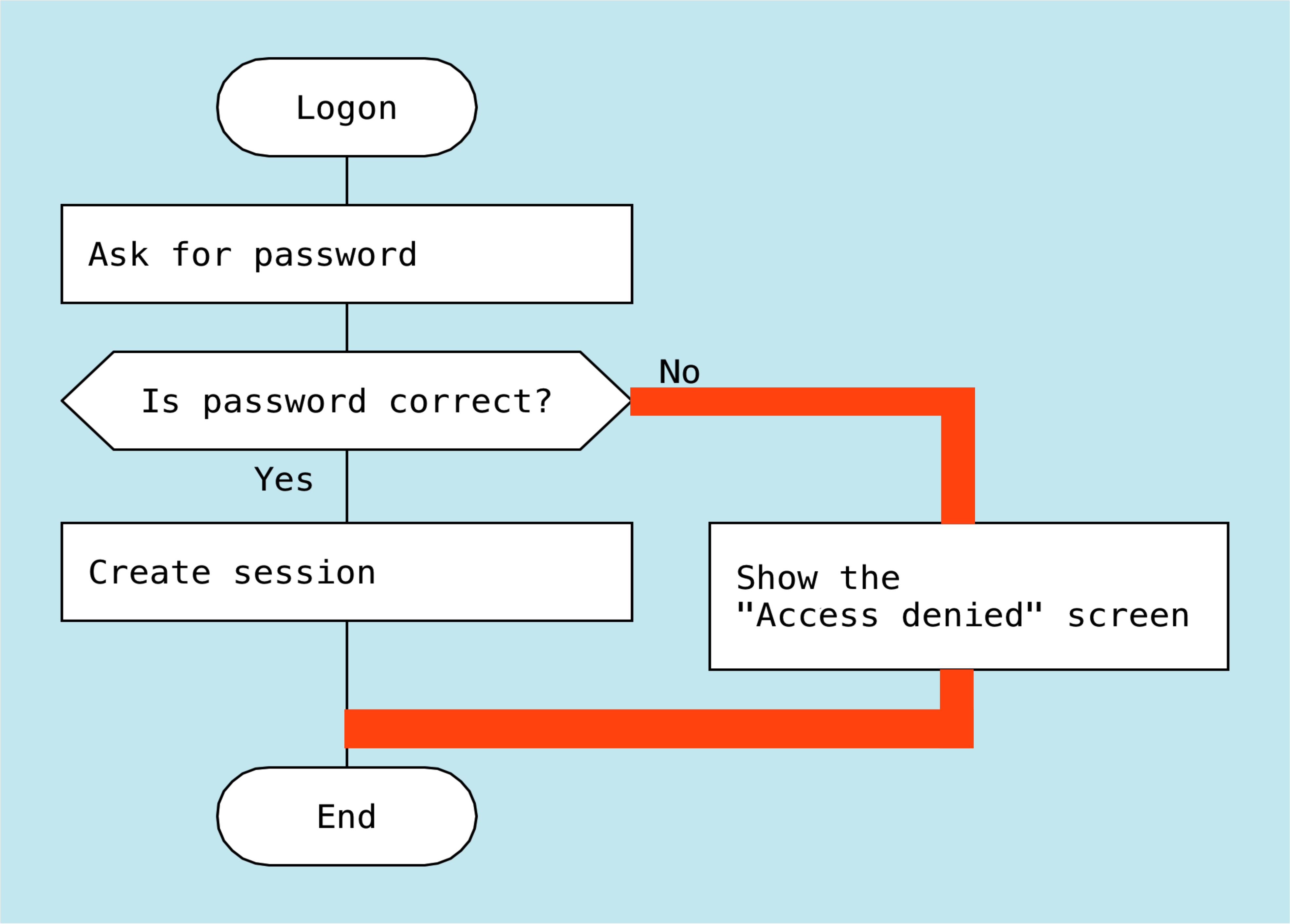
Contains a question
that can be answered
“yes” or “no”

The happy path goes straight

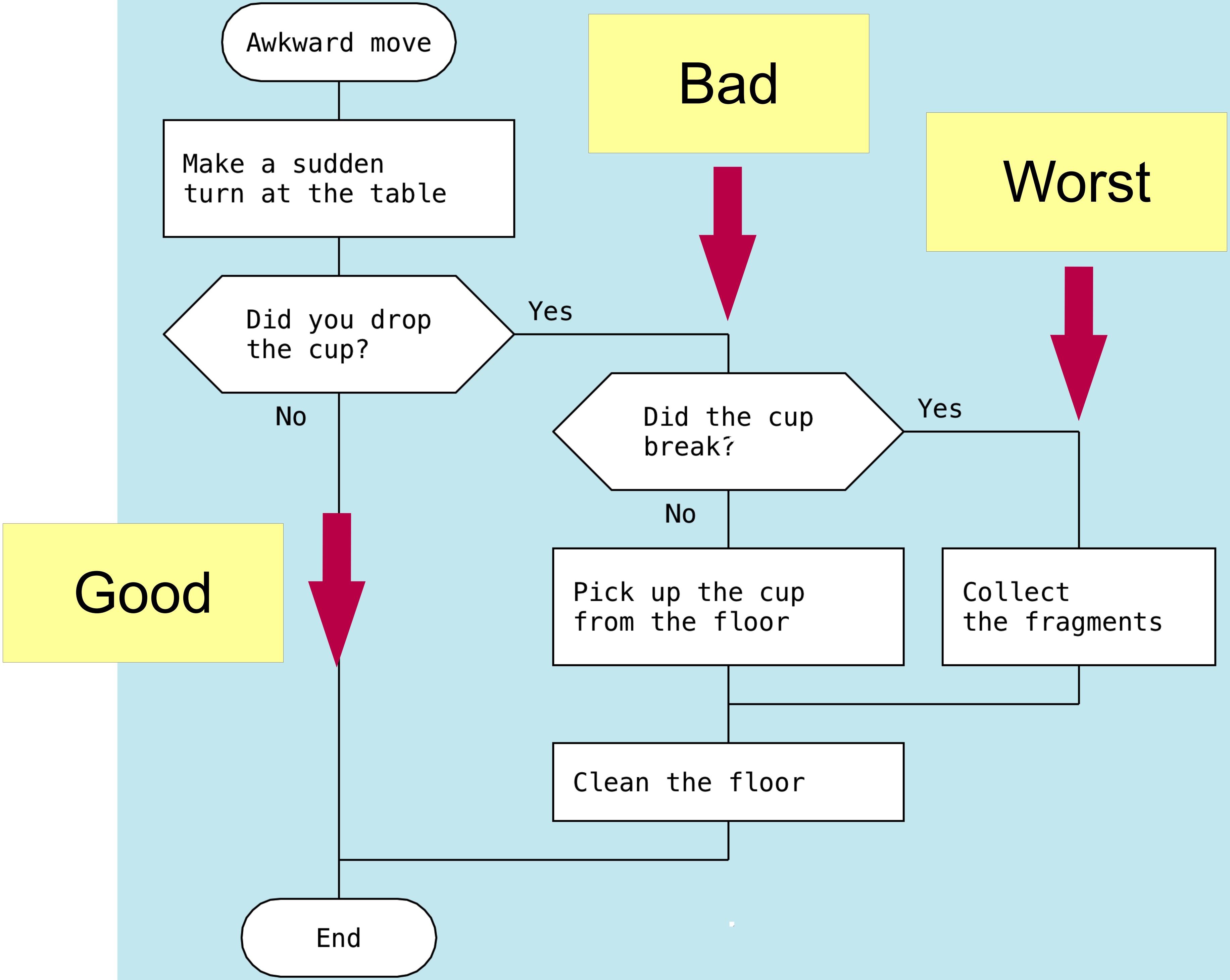


The happy path is the most successful and desirable route

The unlucky scenario goes to the right



The further to the right,
the worse it is



Awkward move

Make a sudden turn at the table

Did you drop the cup?

Yes

No

Did the cup break?

Yes

No

Pick up the cup from the floor

Collect the fragments

Clean the floor

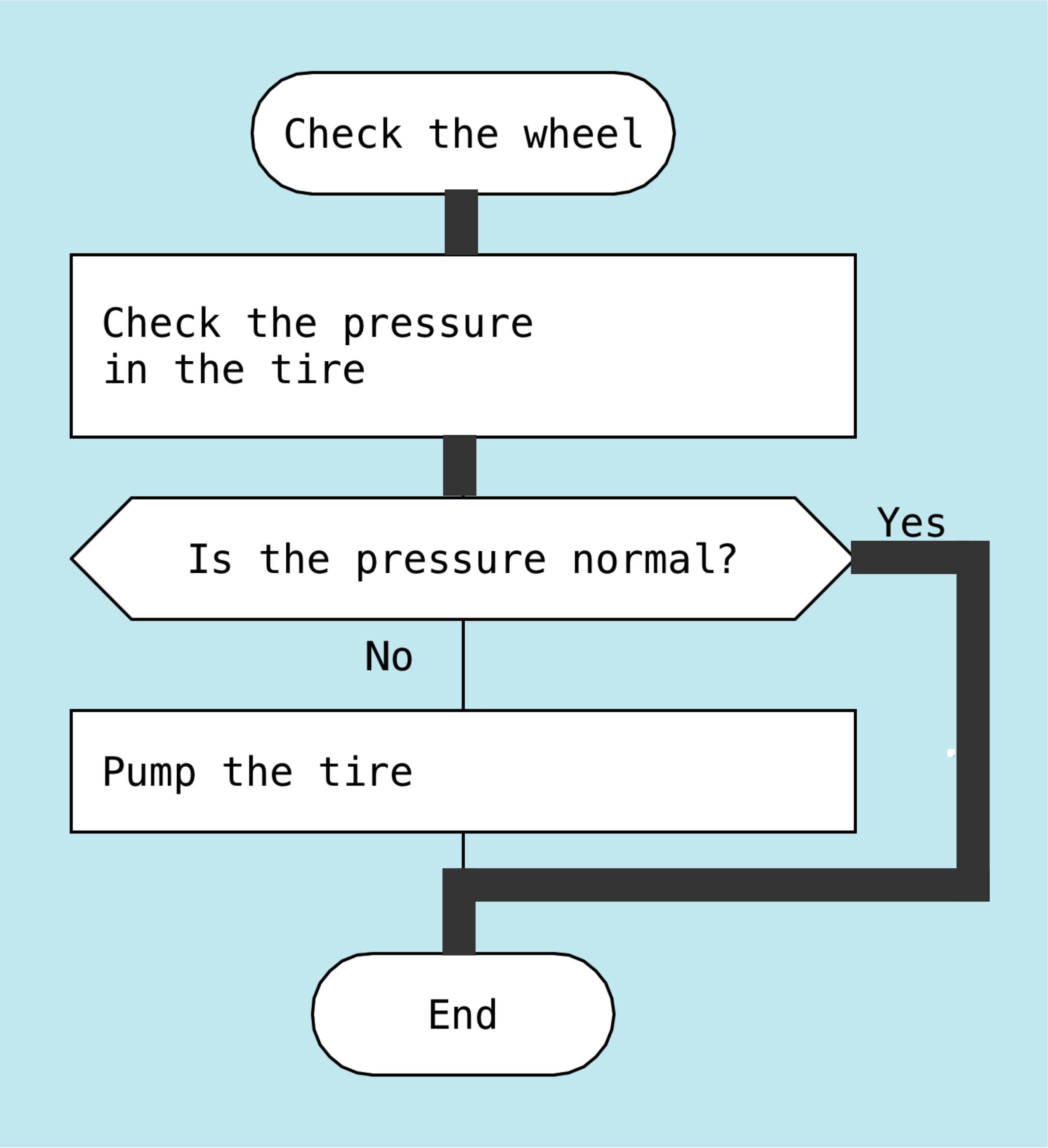
End

Bad

Worst

Good

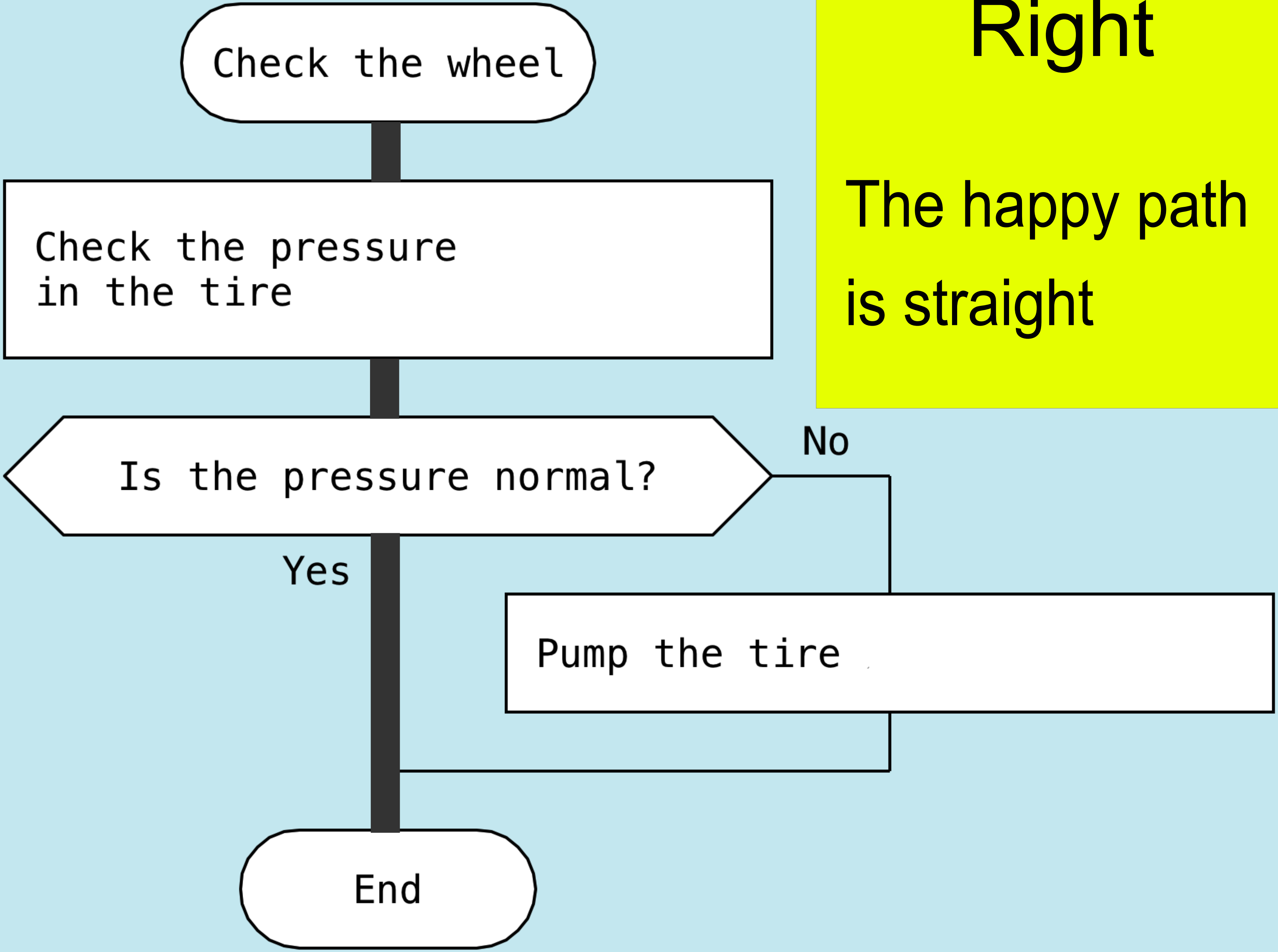
Wrong



The happy path is broken

Right

The happy path is straight



Check the wheel

Check the pressure in the tire

Is the pressure normal?

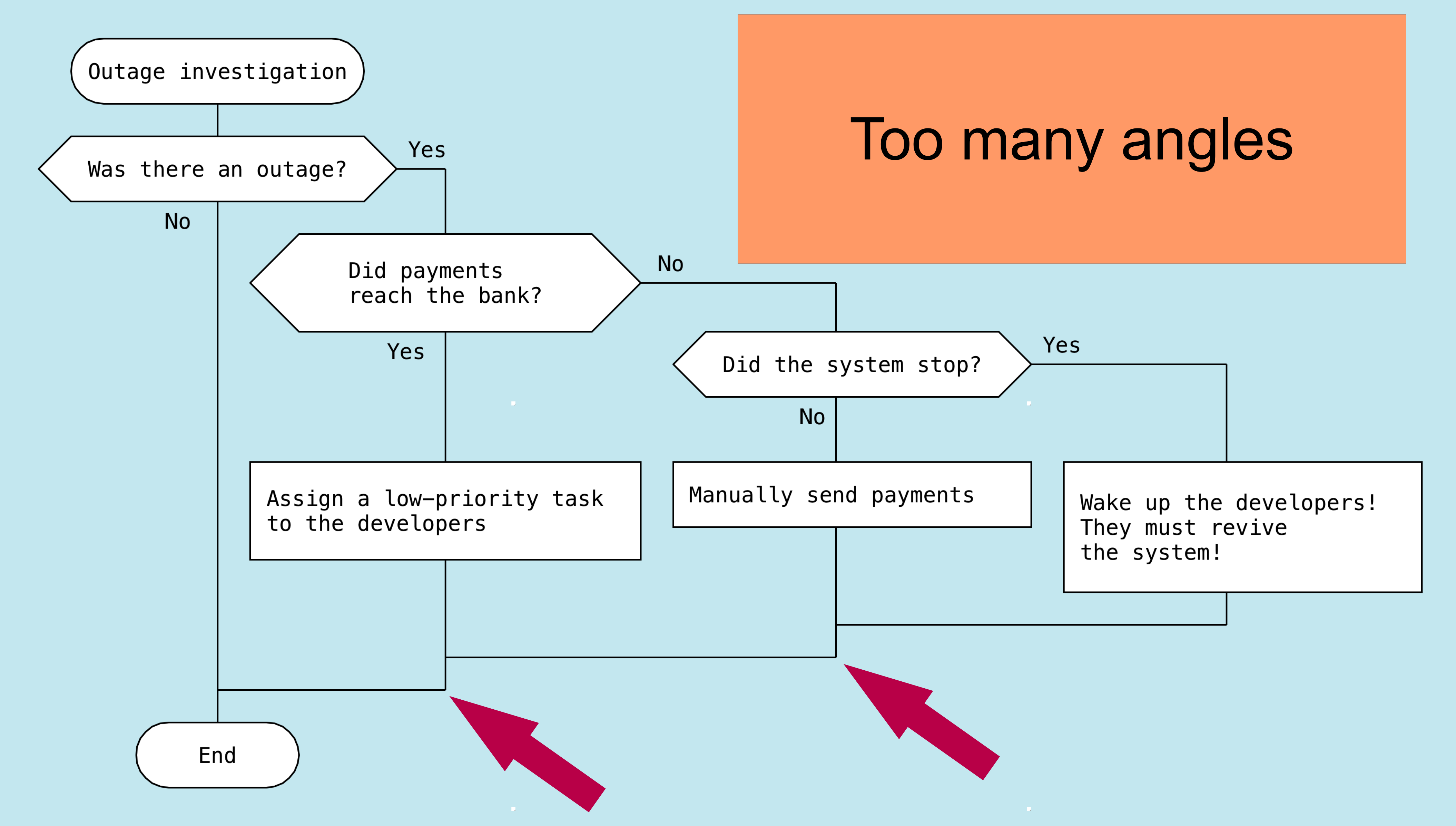
No

Yes

Pump the tire

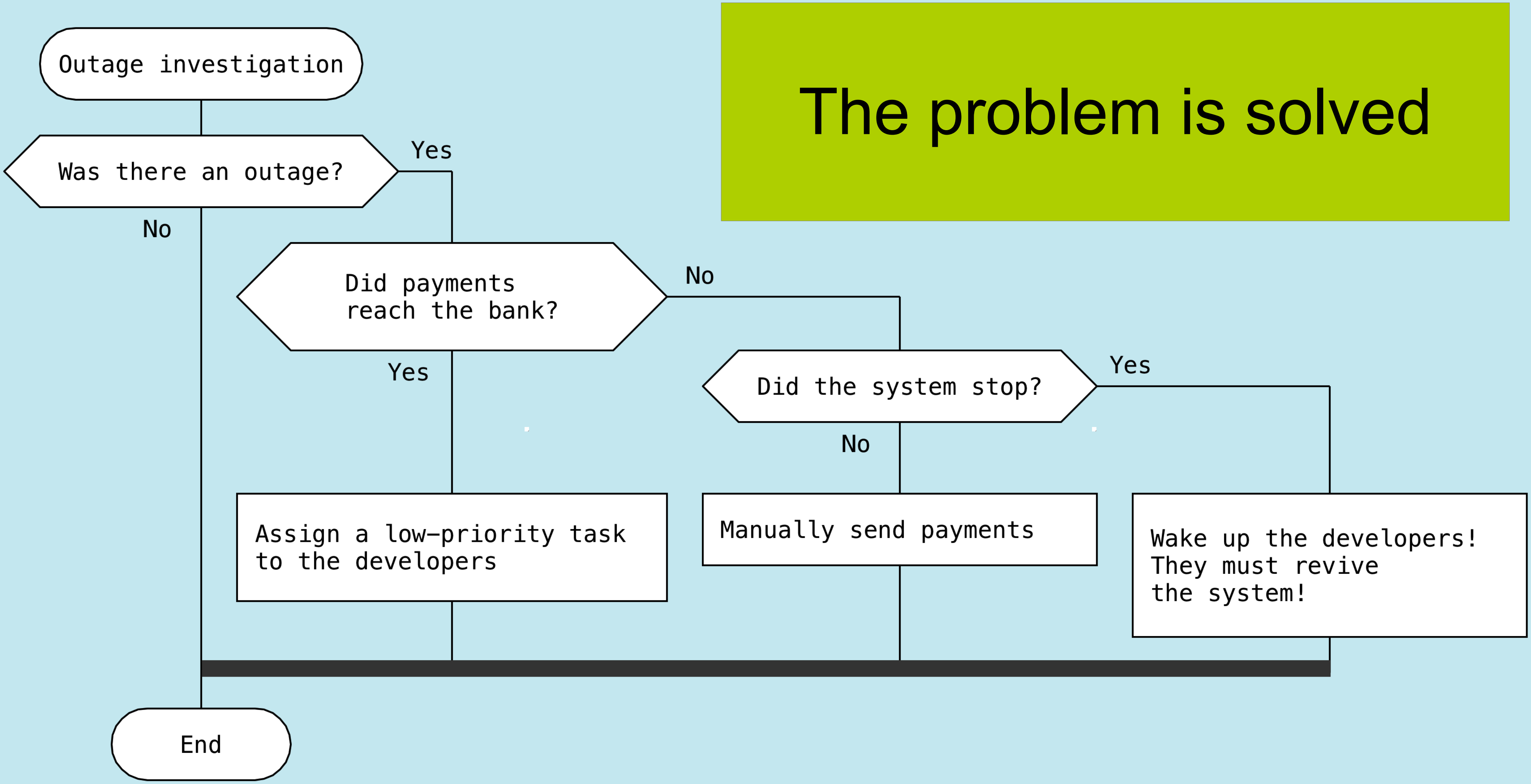
End

Visual noise is bad


Outage investigation
Was there an outage?
Yes
No
Did payments reach the bank?
No
Yes
Did the system stop?
Yes
No
Assign a low-priority task to the developers
Manually send payments
Wake up the developers! They must revive the system!
End

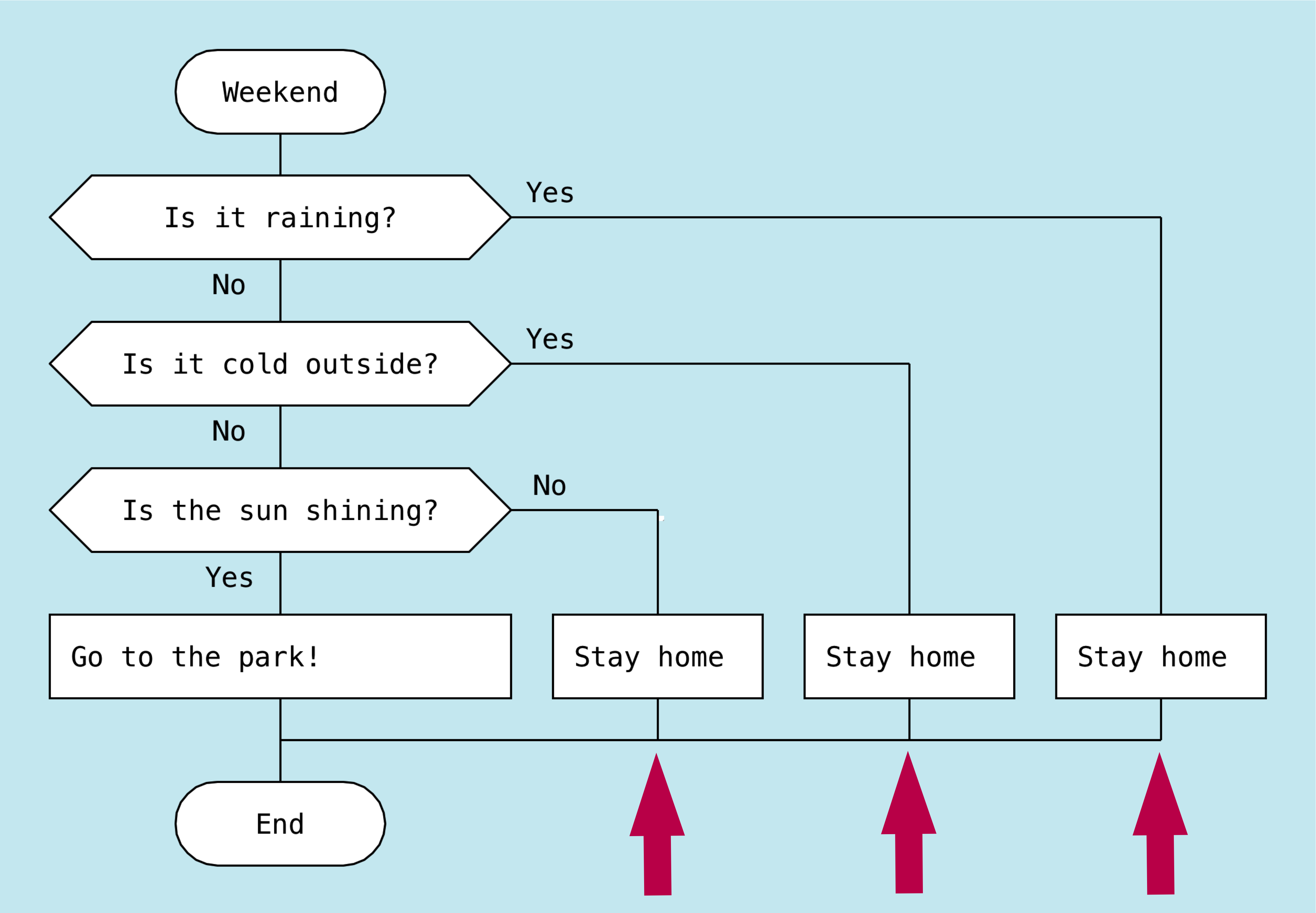
Too many angles

Horizontal joining

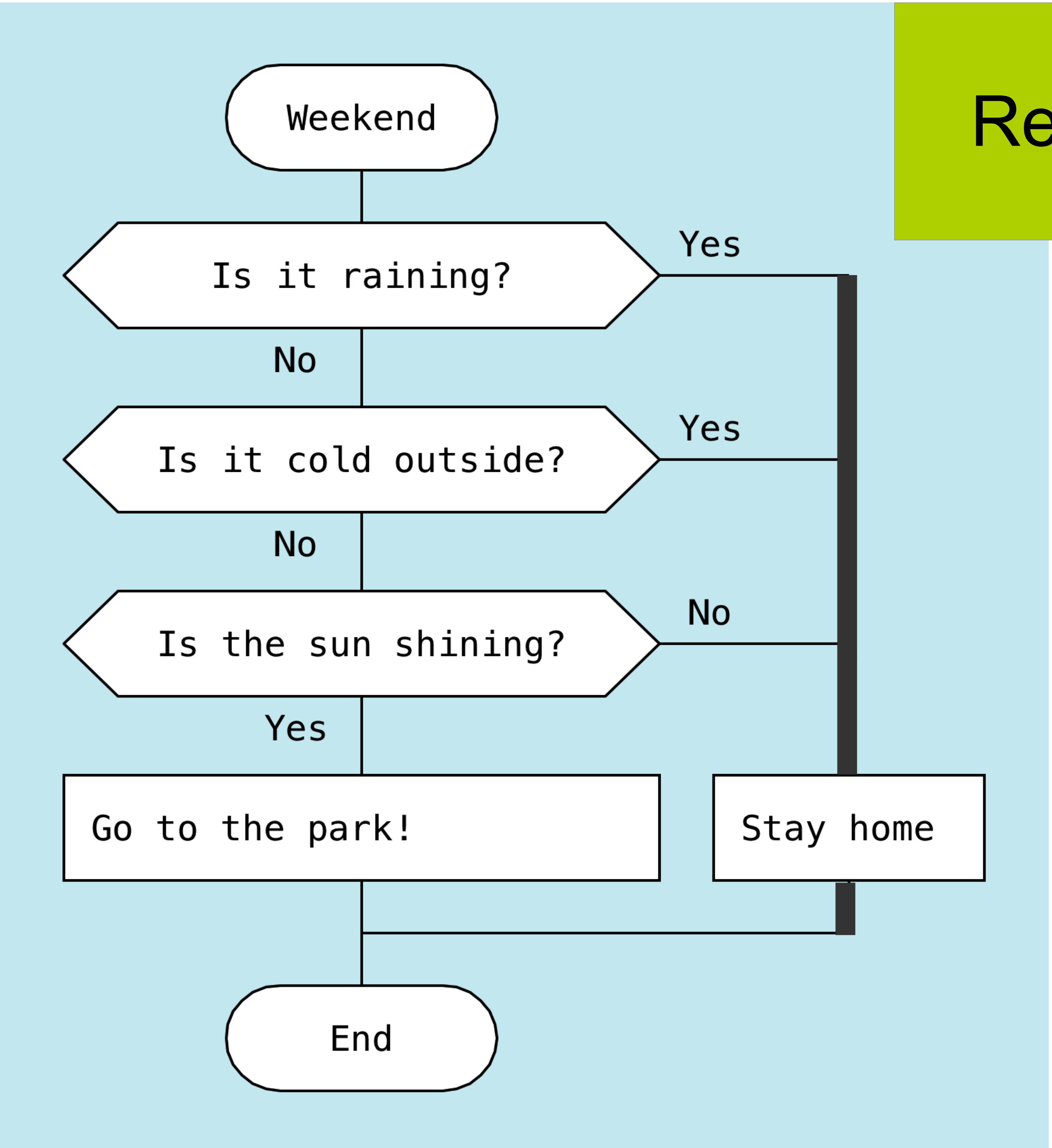

Outage investigation
Was there an outage?
Yes
No
Did payments reach the bank?
No
Yes
Did the system stop?
Yes
No
Assign a low-priority task to the developers
Manually send payments
Wake up the developers! They must revive the system!
End

The problem is solved

Repetition is bad



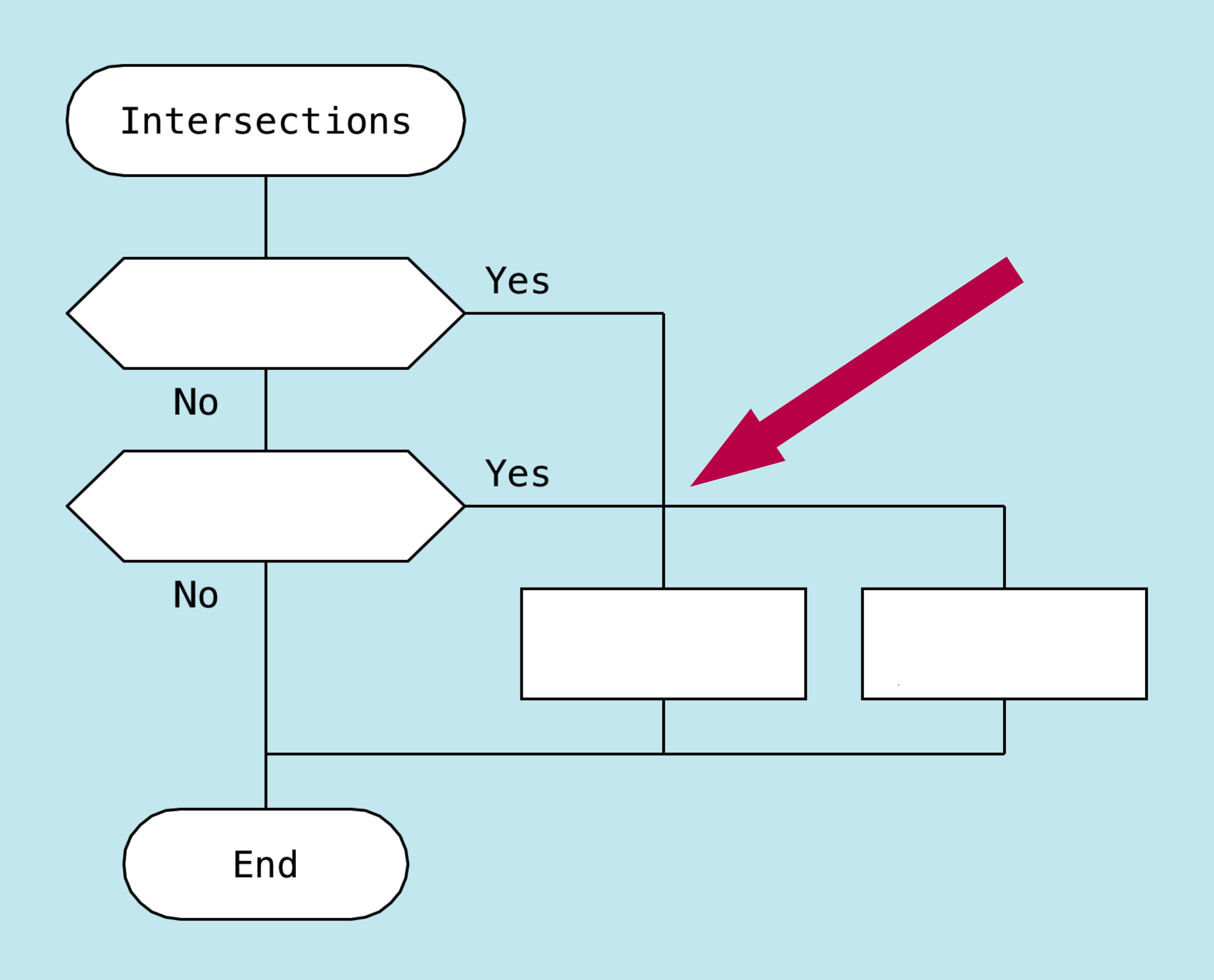
Vertical joining



Removes repetition

Line intersections are forbidden

Intersections kill readability



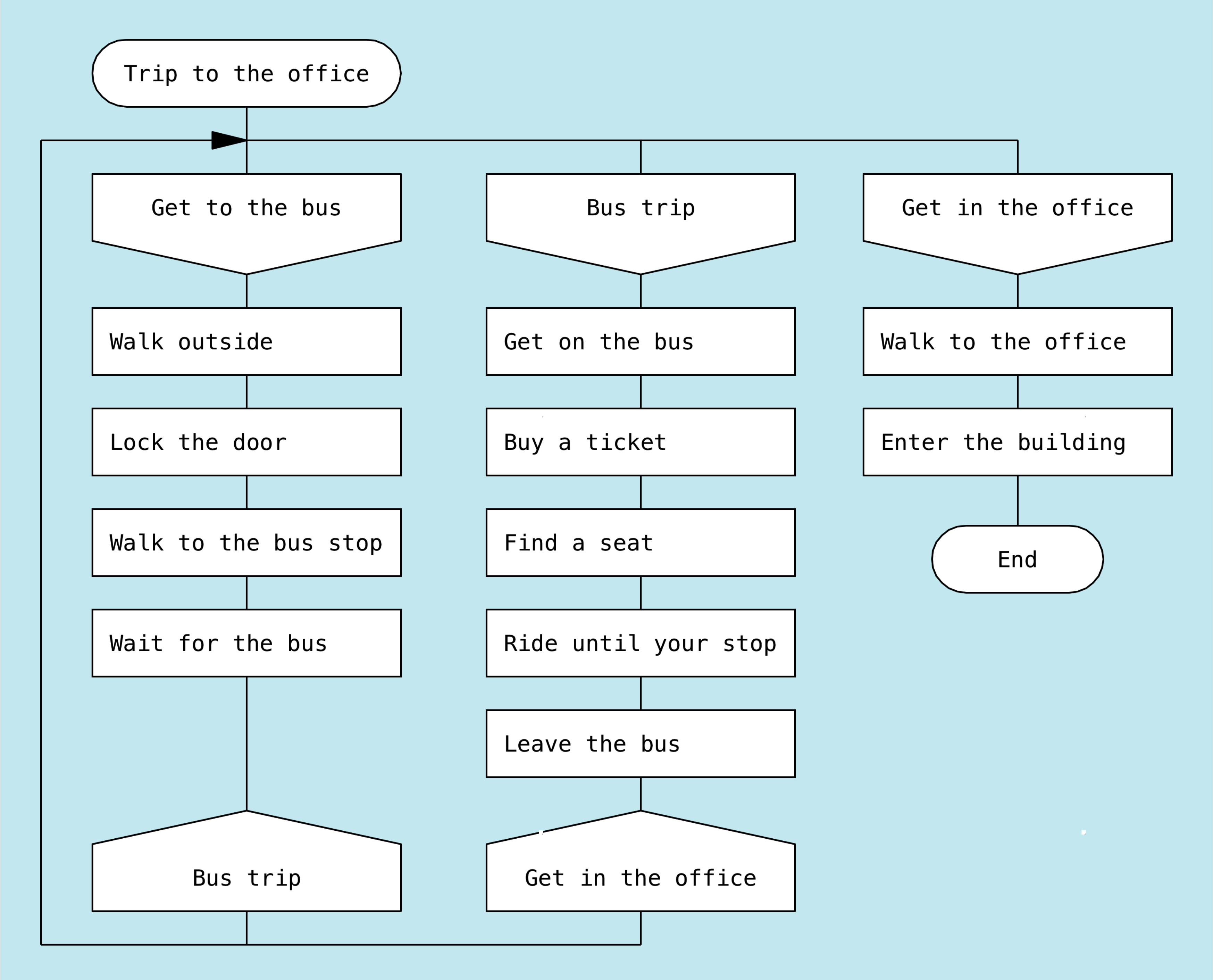
Silhouette

The recommended type of diagrams

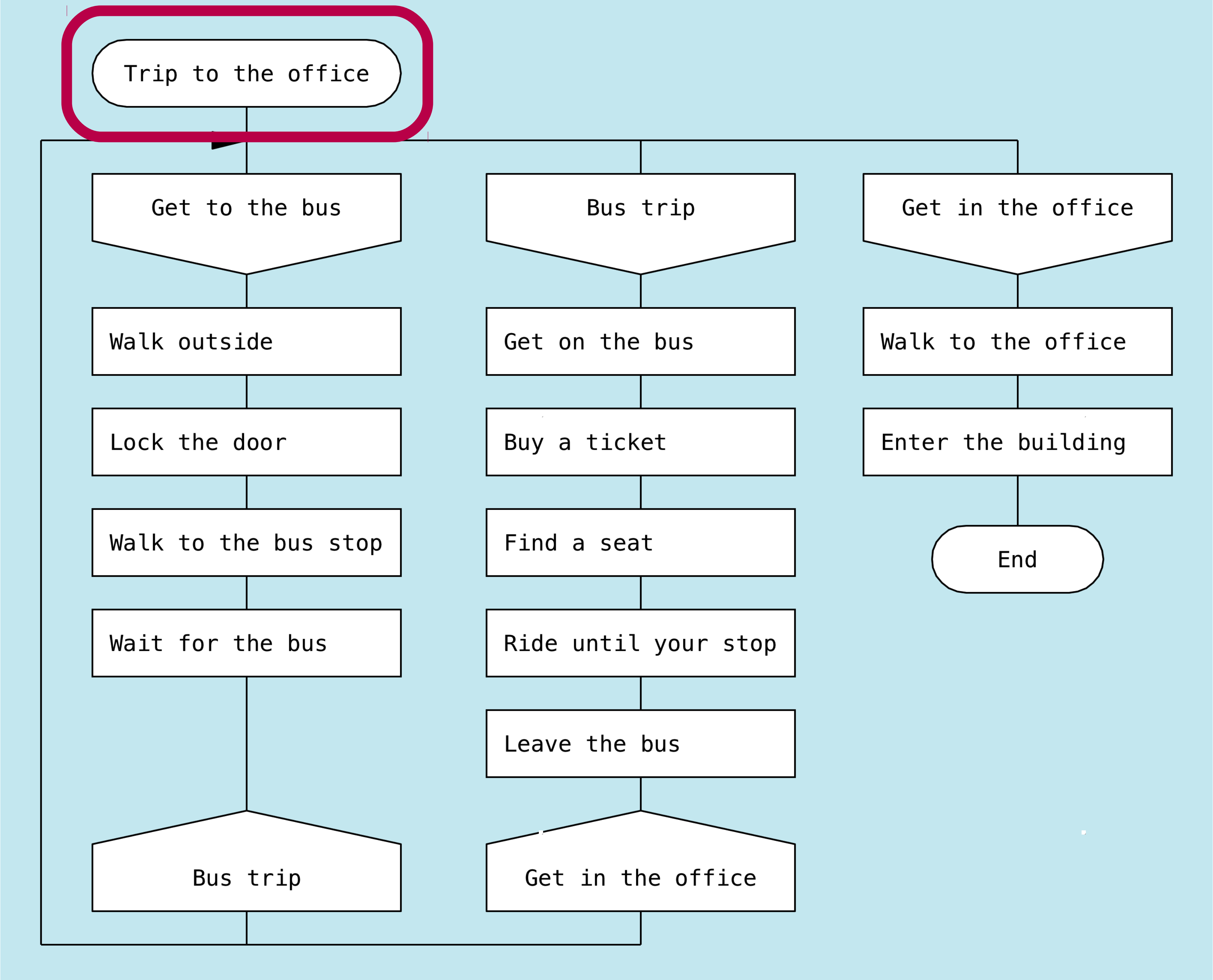
What can silhouette do?

- It can cleanly split the diagram into logical parts
- It can represent a finite automaton (state machine)
- It helps get rid of line intersections in complex algorithms

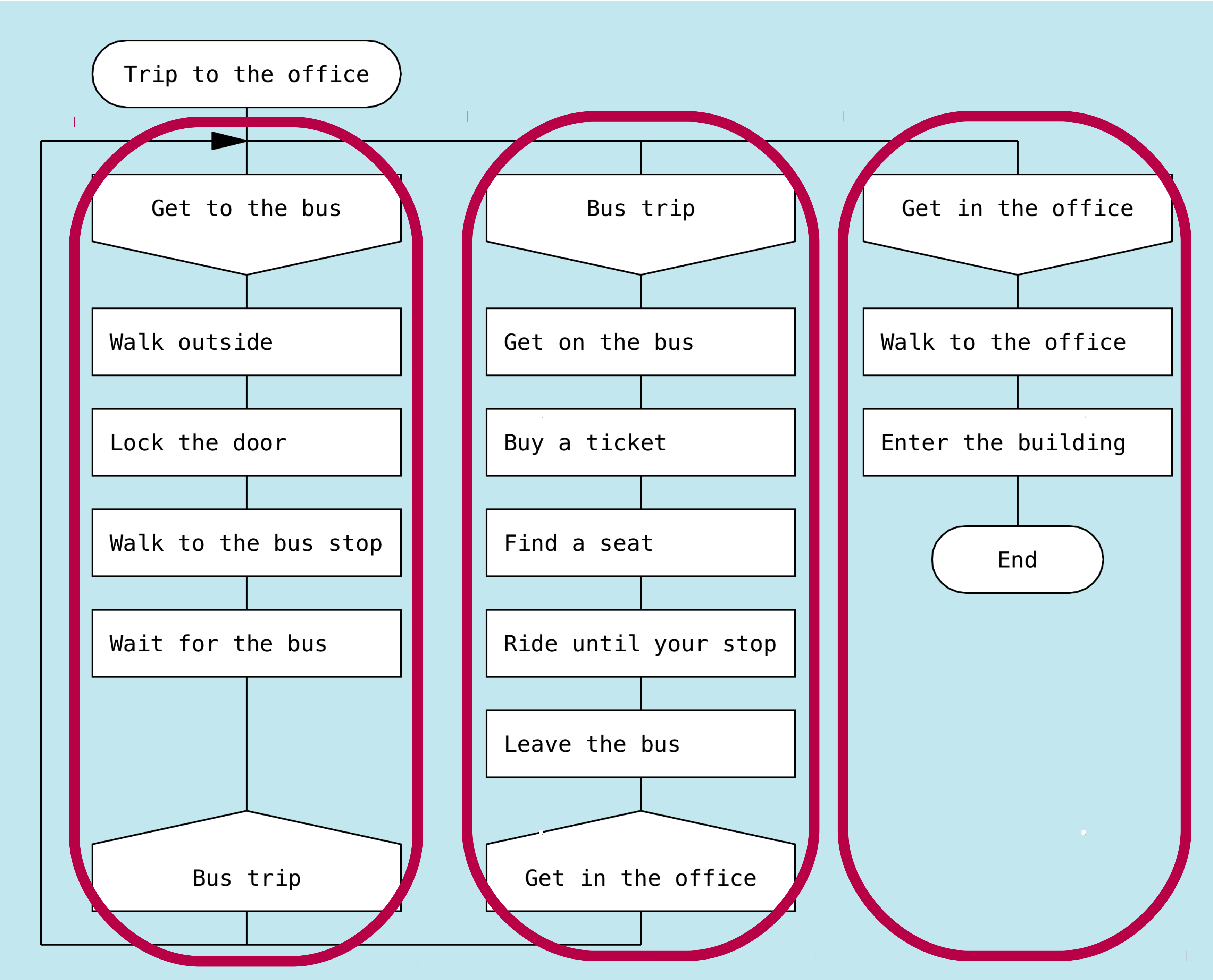
What does a silhouette consist of?



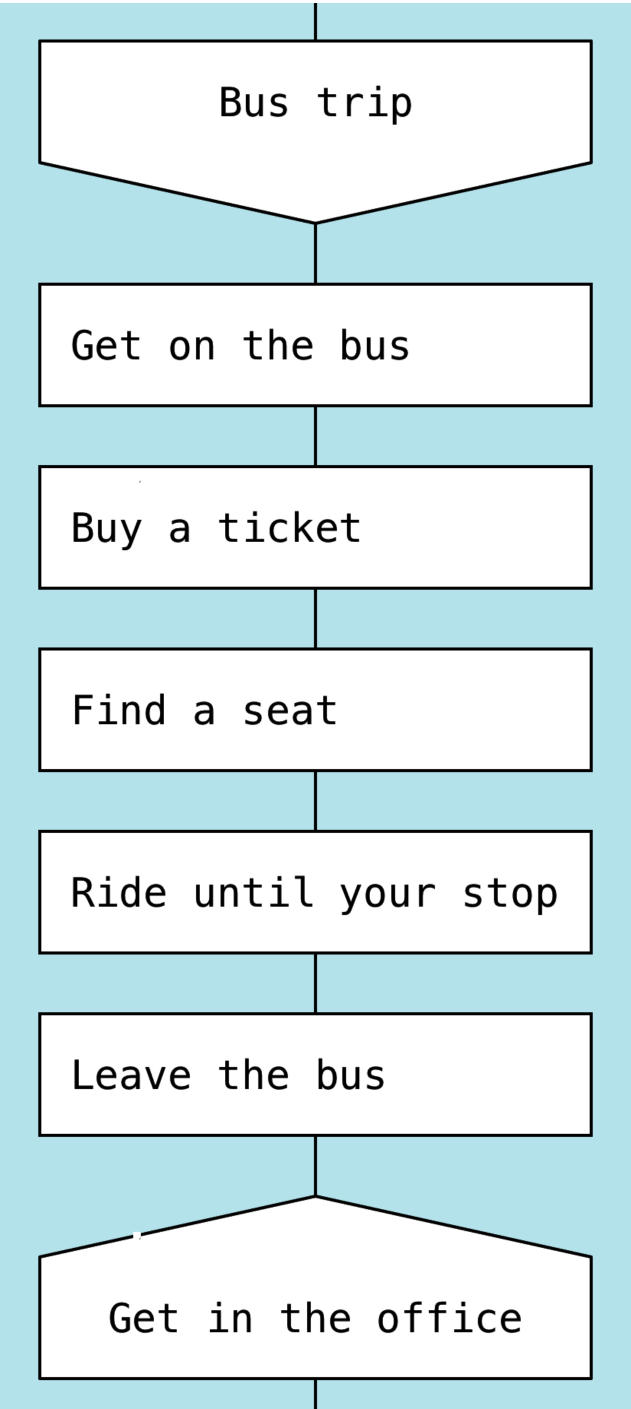
The header


Trip to the office
Get to the bus
Bus trip
Get in the office
Walk outside
Get on the bus
Walk to the office
Lock the door
Buy a ticket
Enter the building
Walk to the bus stop
Find a seat
End
Wait for the bus
Ride until your stop
Leave the bus
Bus trip
Get in the office

The branches



What does a branch consist of?



Bus trip

Get on the bus

Buy a ticket

Find a seat

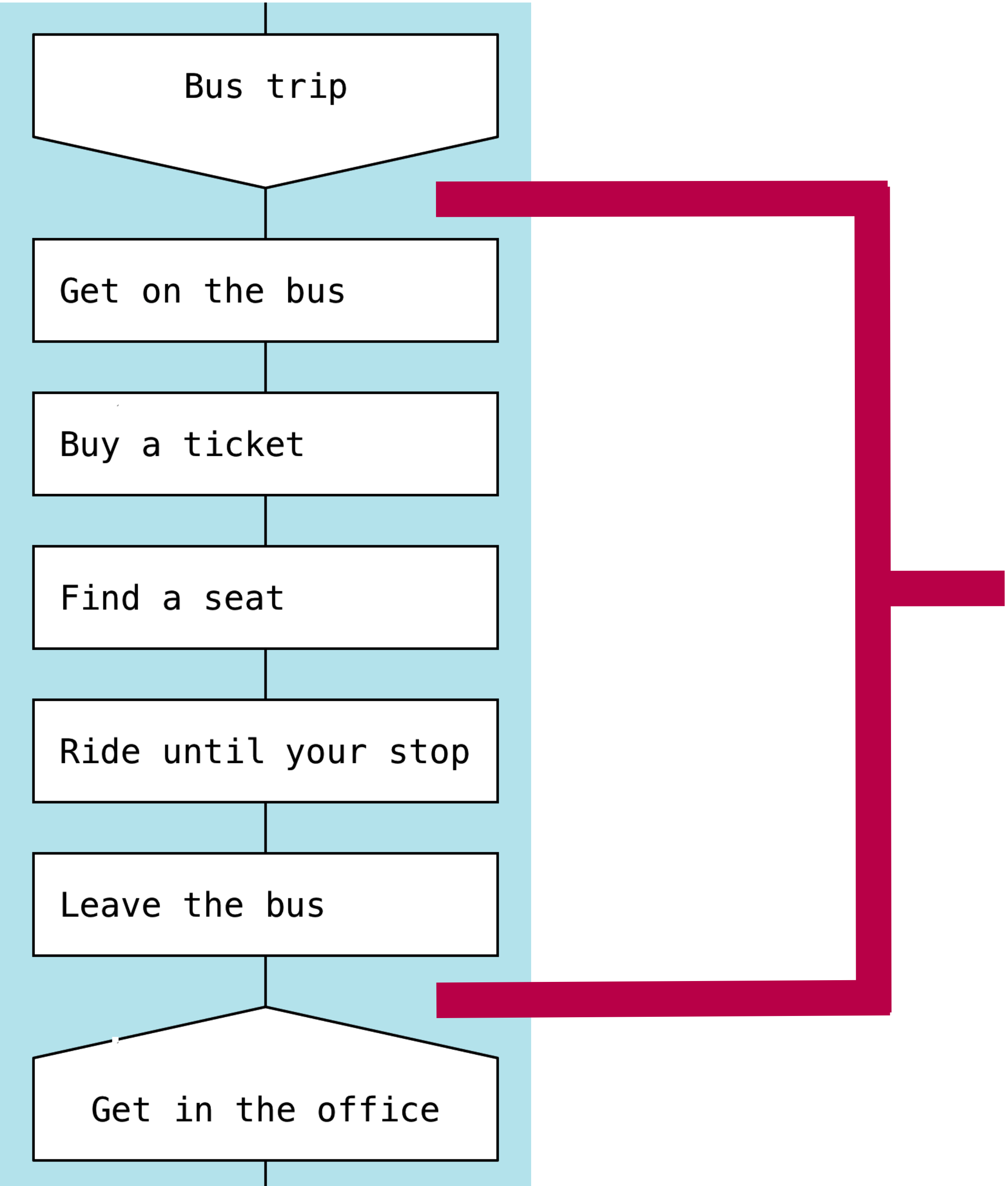
Ride until your stop

Leave the bus

Get in the office

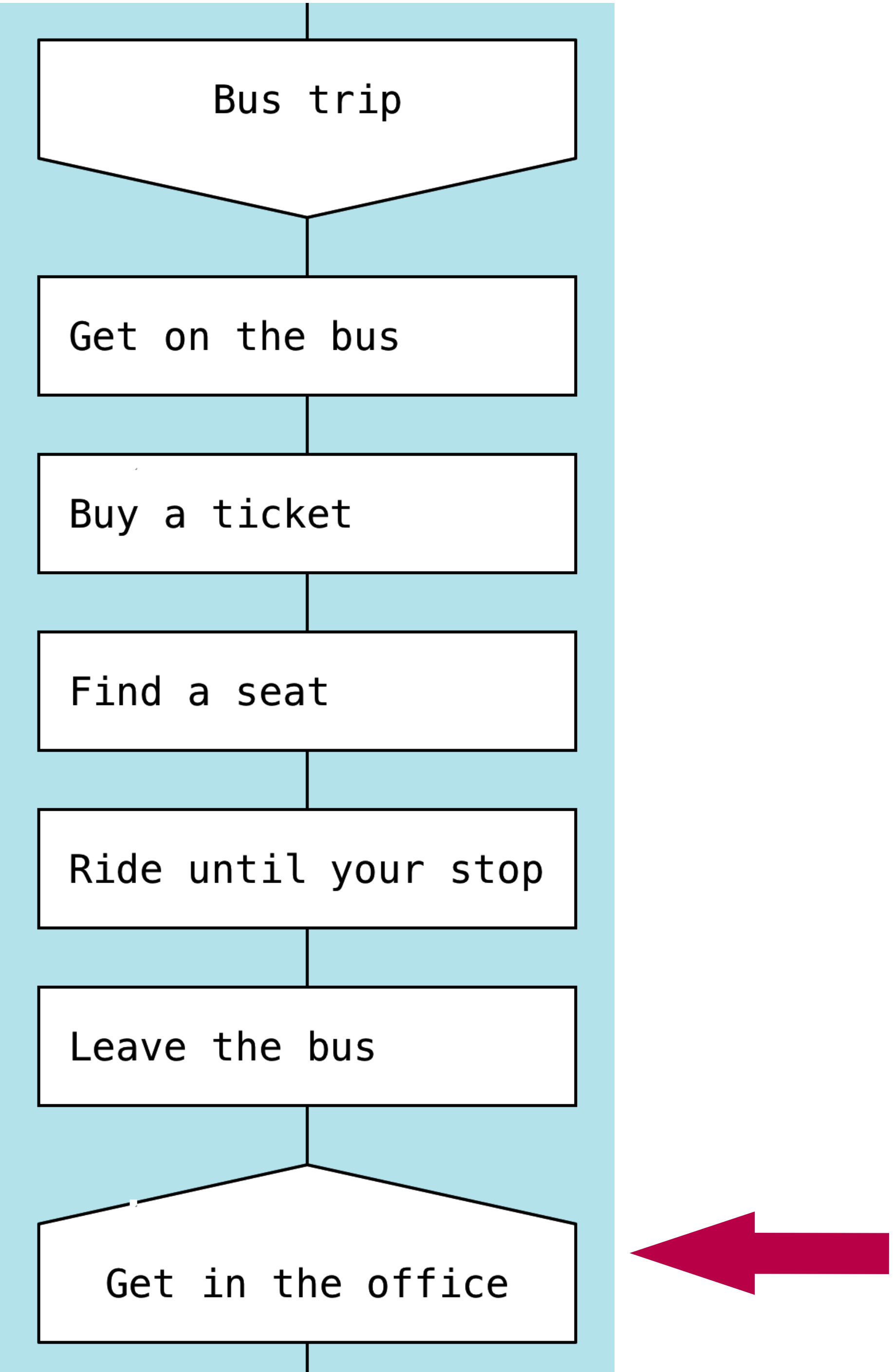
A header with
the branch name

What does a branch consist of?



The algorithm of the branch

What does a branch consist of?

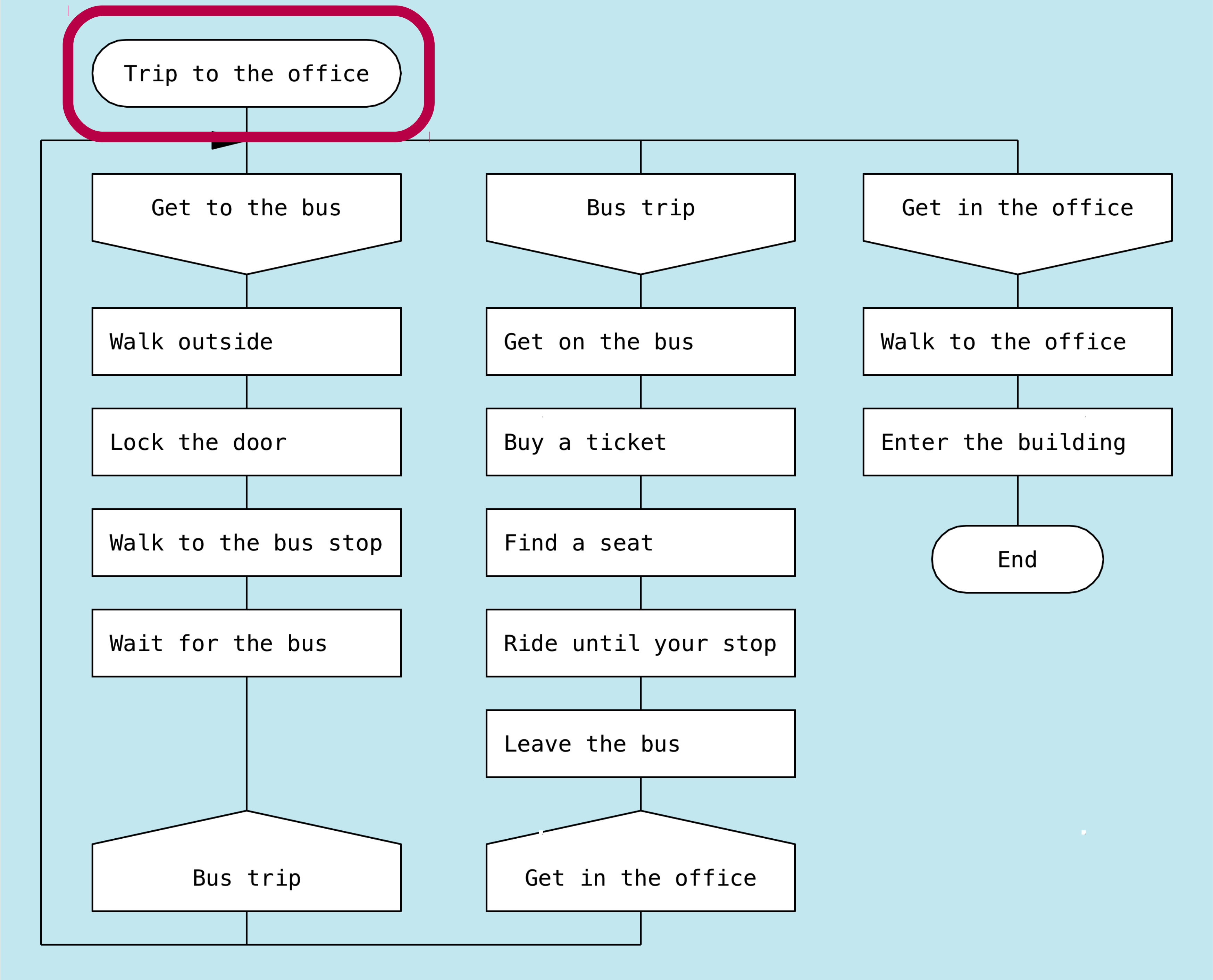


The address
of the next branch

Silhouette answers the three questions of the king:

1. What is the name of the problem?
2. How many parts does the problem have?
3. What are the names of the parts?

1. What is the name of the problem?



Trip to the office

Get to the bus

Walk outside

Lock the door

Walk to the bus stop

Wait for the bus

Bus trip

Bus trip

Get on the bus

Buy a ticket

Find a seat

Ride until your stop

Leave the bus

Get in the office

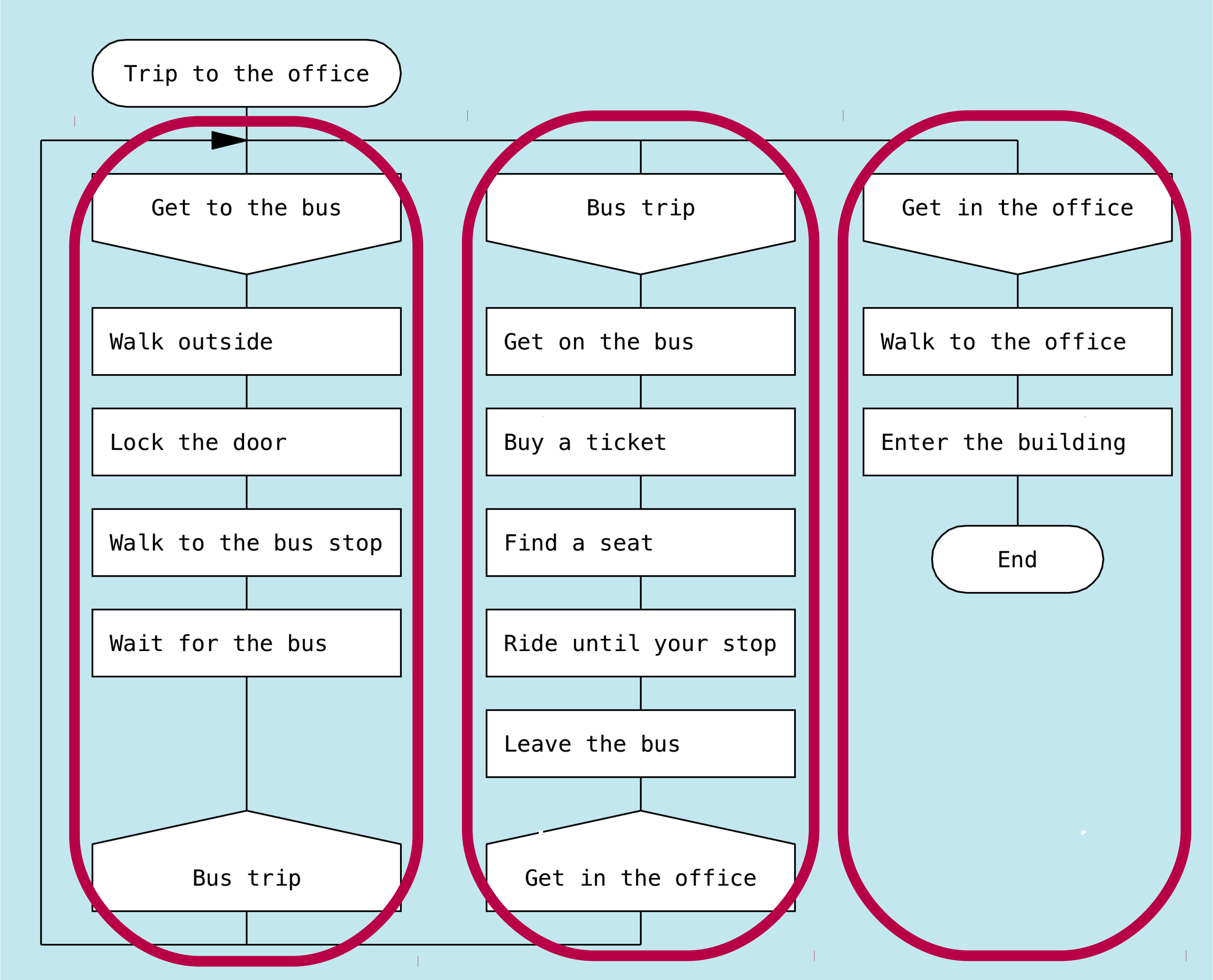
Get in the office

Walk to the office

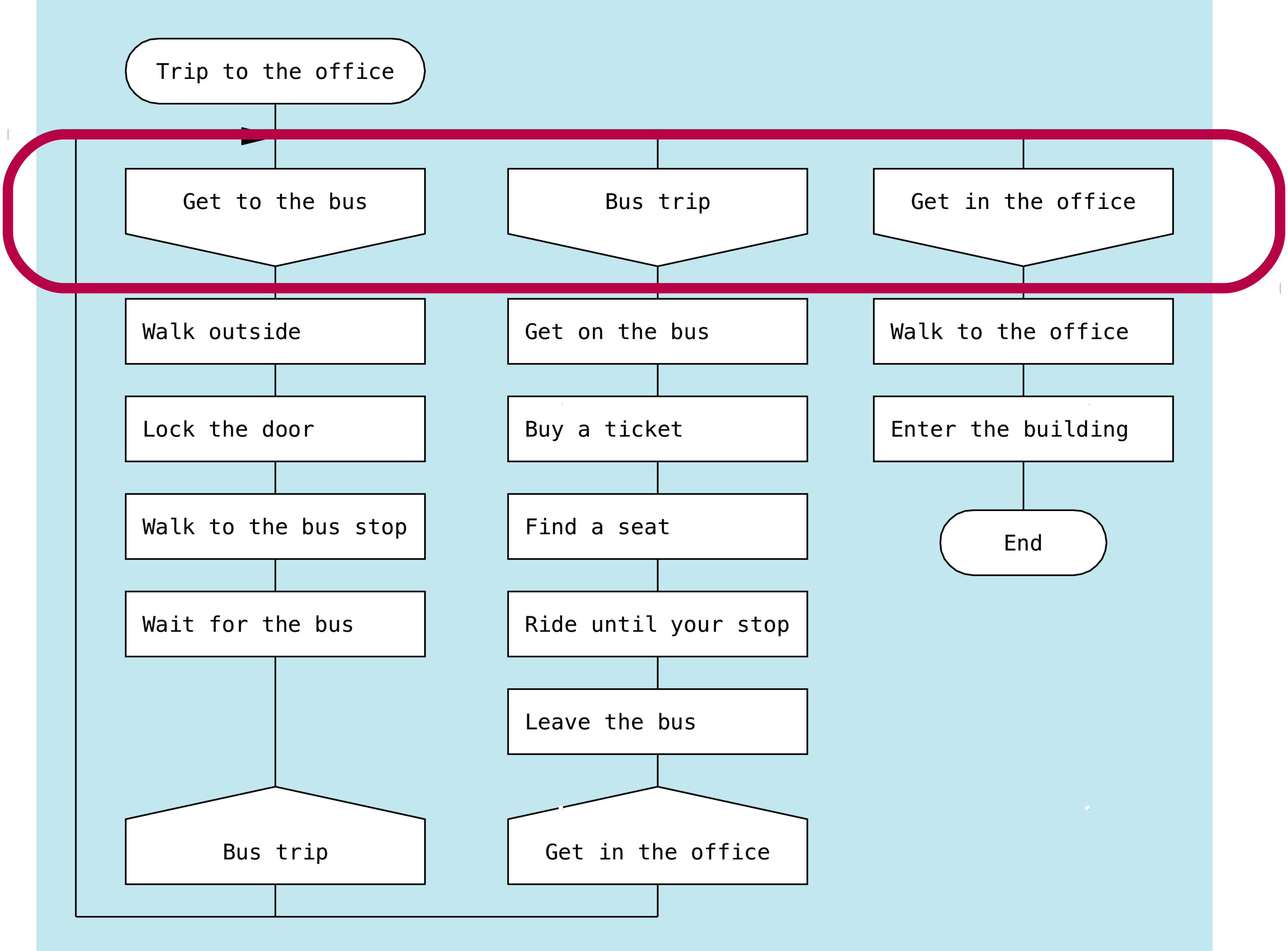
Enter the building

End

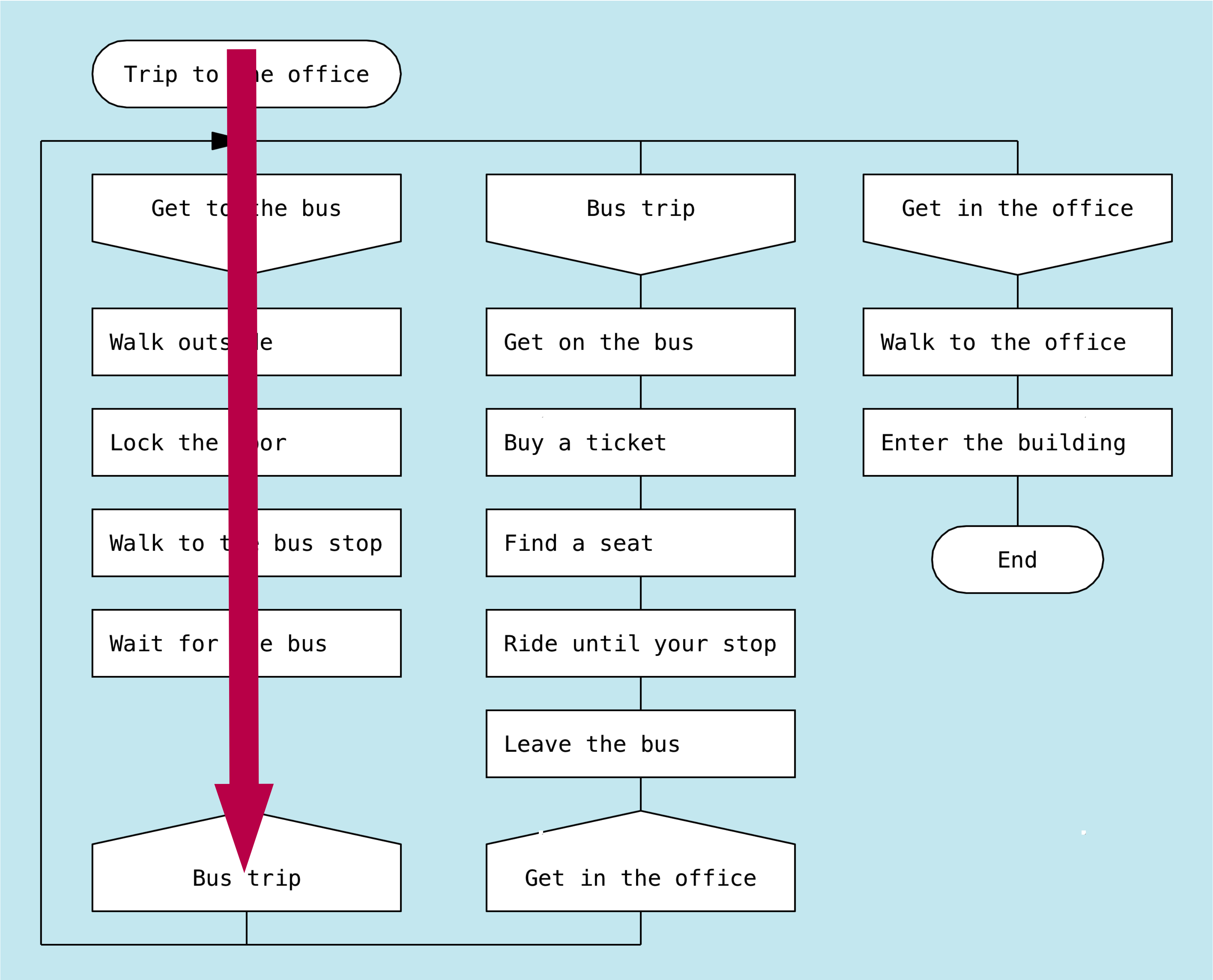
2. How many problems does the problem have?



3. What are the names of the parts?



The leftmost branch is run first



Trip to the office

Get to the bus

Walk outside

Lock the door

Walk to the bus stop

Wait for the bus

Bus trip

Bus trip

Get on the bus

Buy a ticket

Find a seat

Ride until your stop

Leave the bus

Get in the office

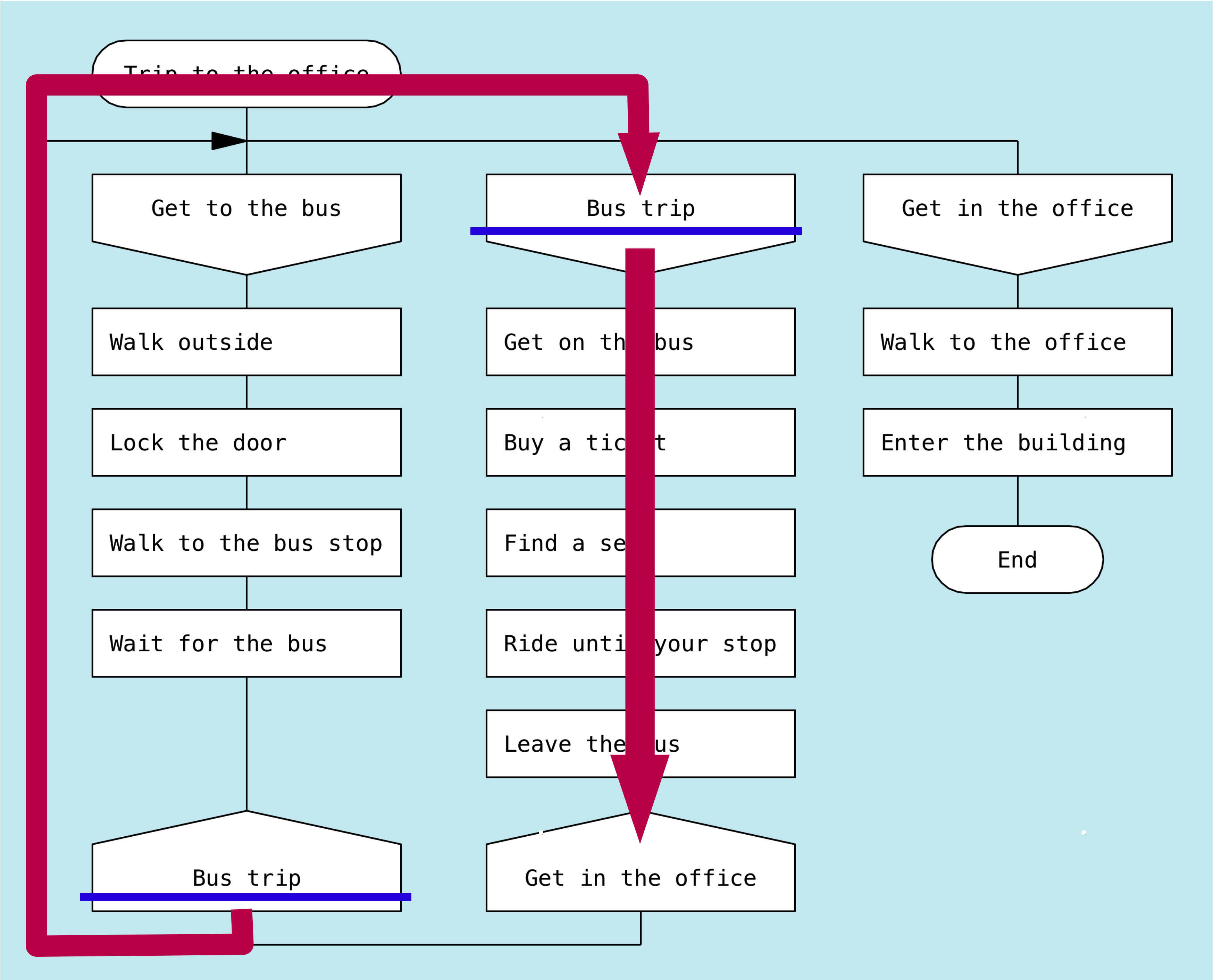
Get in the office

Walk to the office

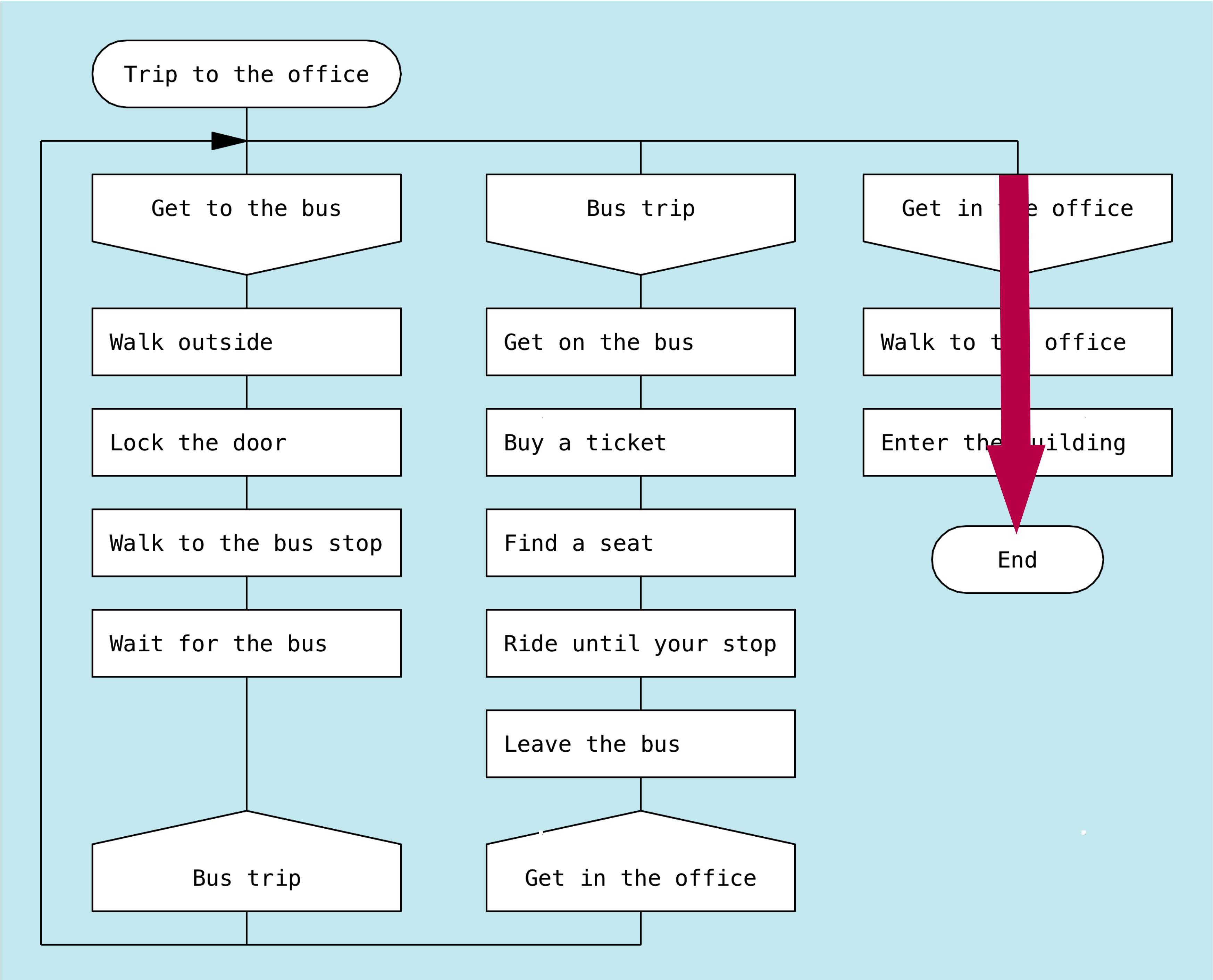
Enter the building

End

Then follow the other branches



The rightmost branch goes last



Trip to the office

Get to the bus

Walk outside

Lock the door

Walk to the bus stop

Wait for the bus

Bus trip

Bus trip

Get on the bus

Buy a ticket

Find a seat

Ride until your stop

Leave the bus

Get in the office

Get in the office

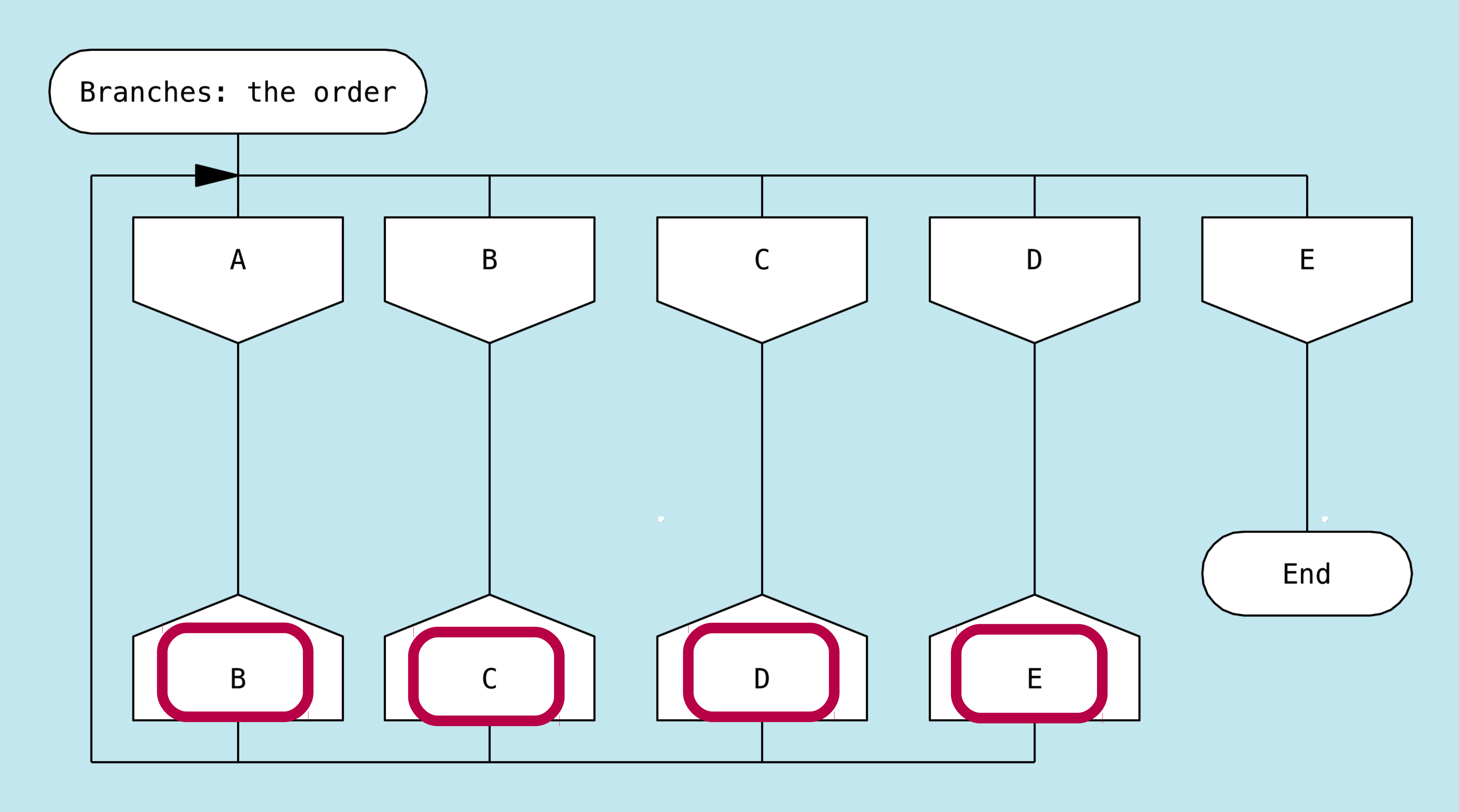
Walk to the office

Enter the building

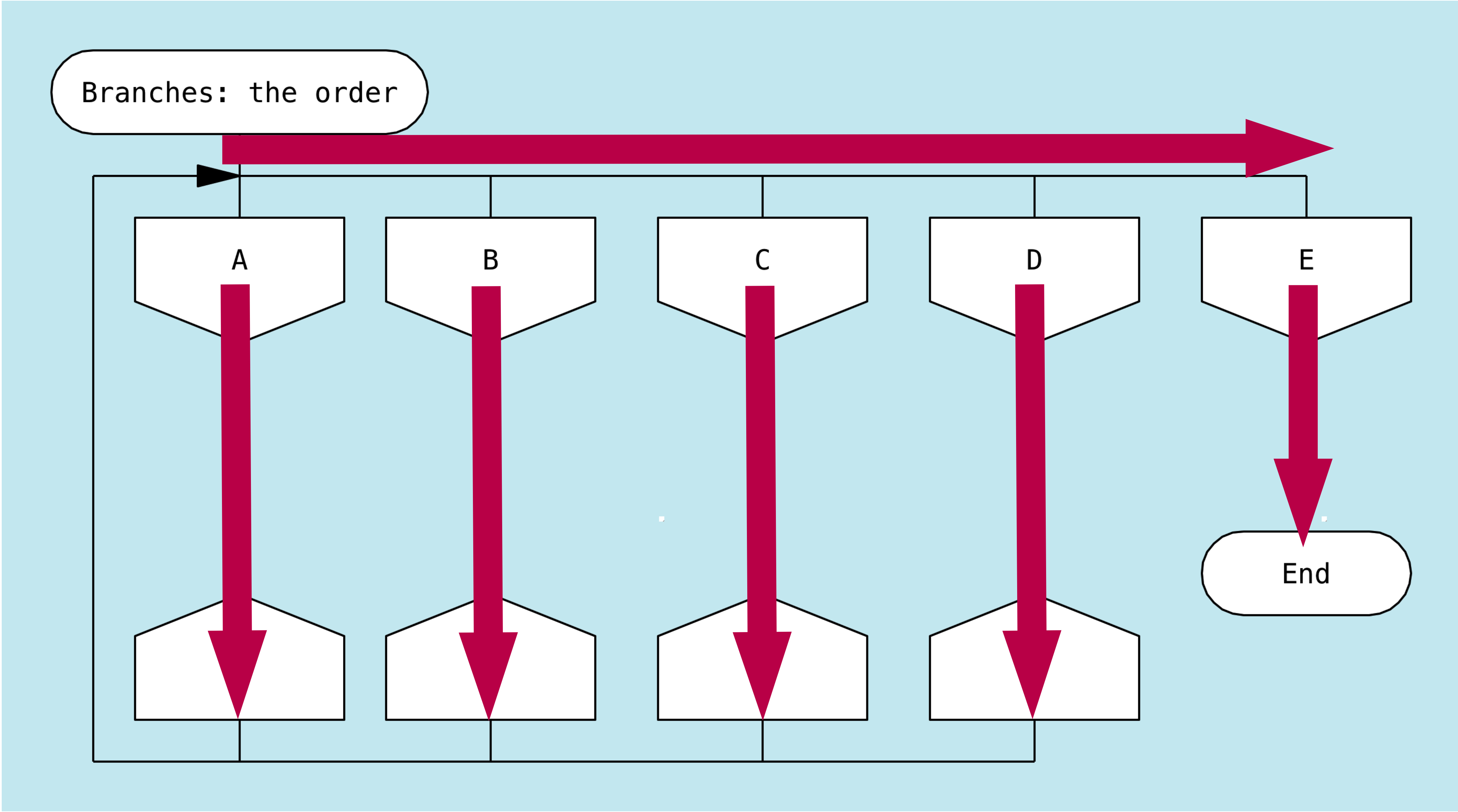
End

There can be only one “End” icon on the DRAKON diagram

The order of branches is defined in the “Address” icons



The branches should by sorted left-to-right

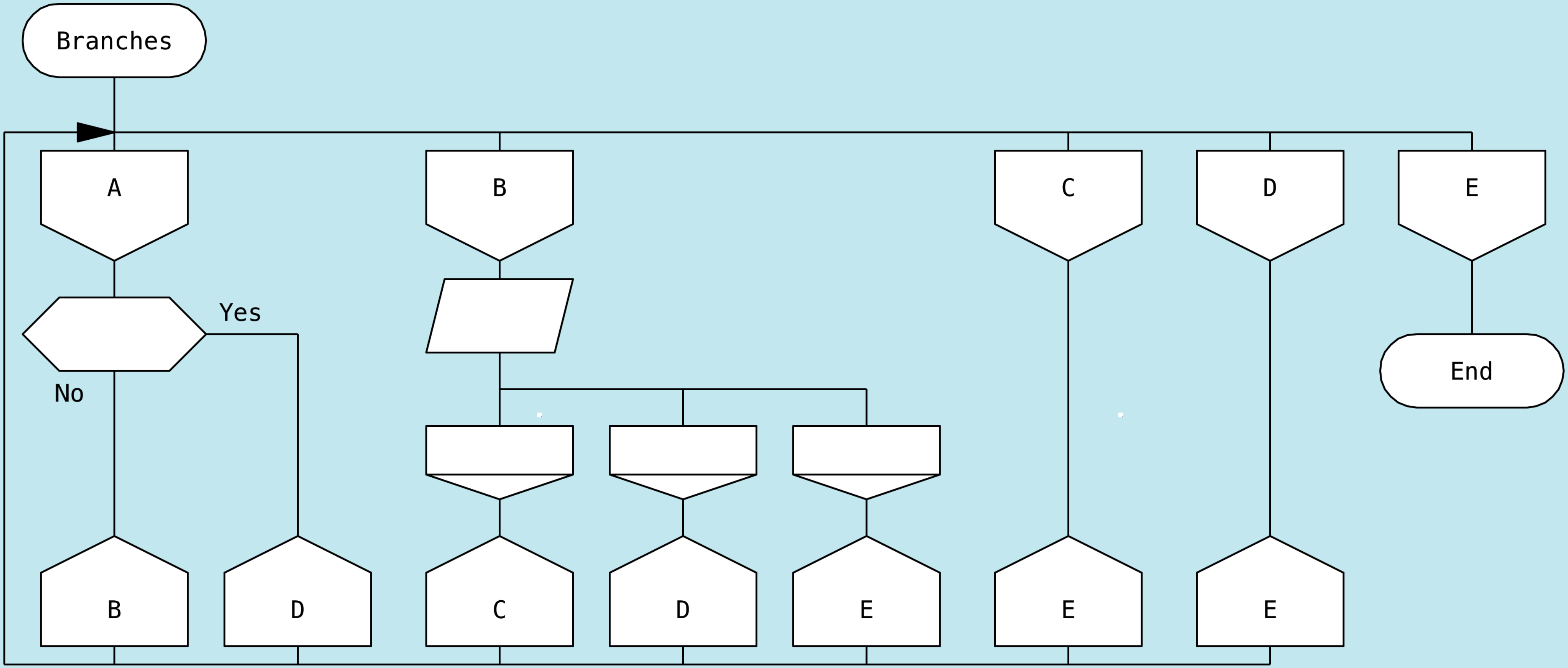


When is it allowed to go to the left?

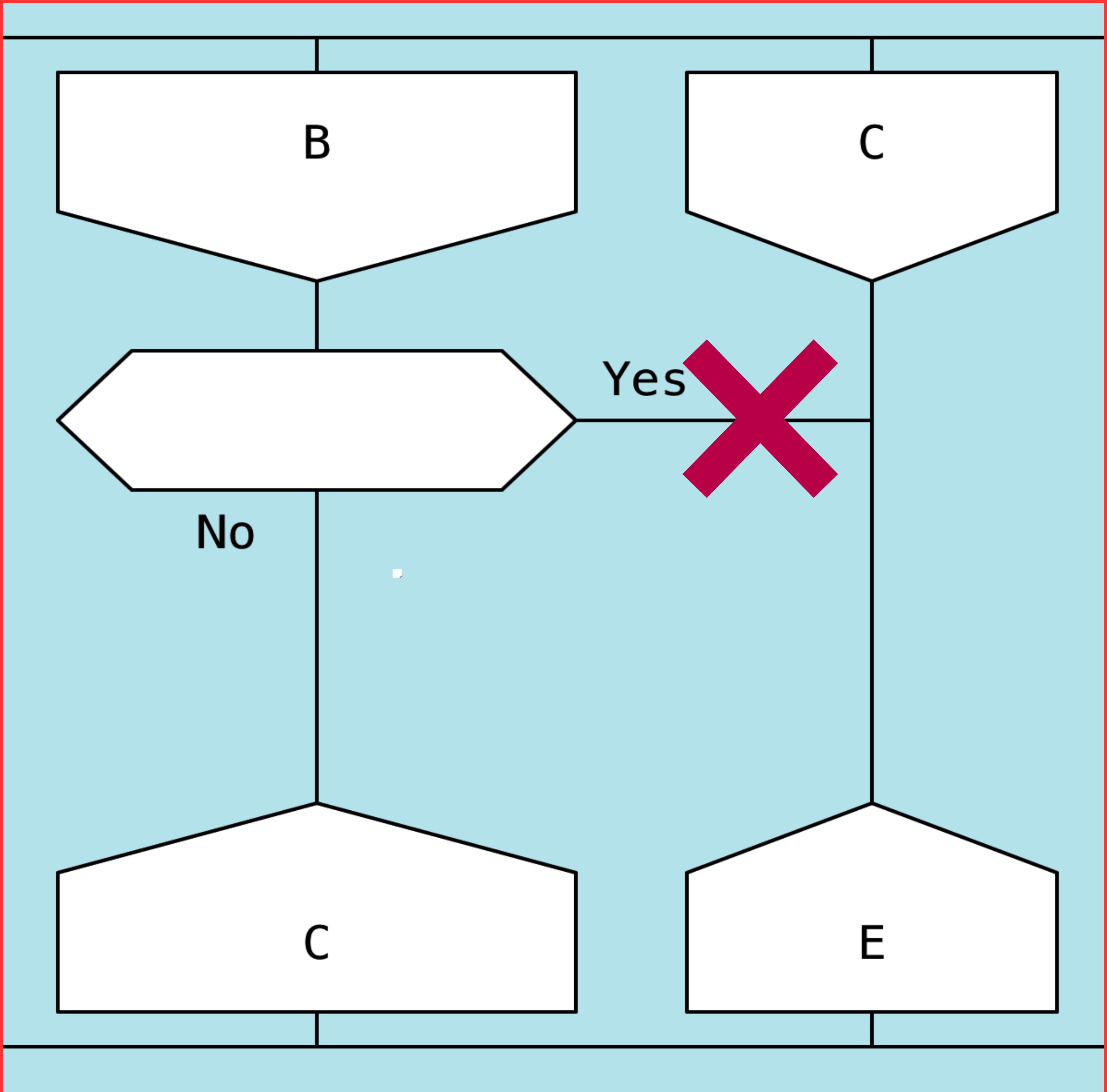
1. When some branches need to be repeated.
2. When the diagram represents a finite automaton (state machine).

A branch can have several “Address” icons

The algorithm of such branch
decides which branch to run next



A direct jump from one branch to another is forbidden

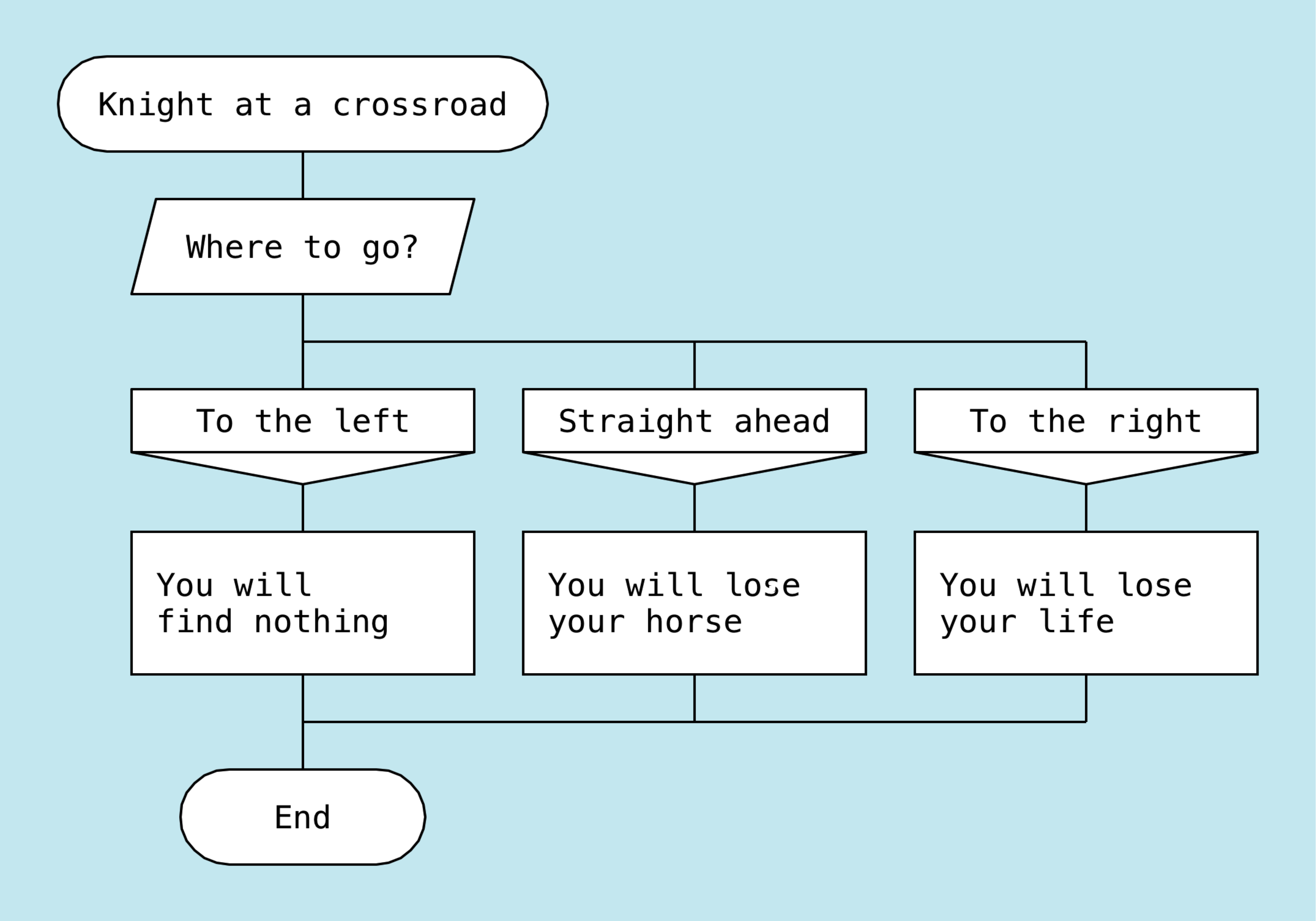


Icon “Select”

Contains a question:

- that cannot be answered “yes” or “no”
- that can have several answers

Icon “Select”


Knight at a crossroad
Where to go?
To the left
Straight ahead
To the right
You will find nothing
You will lose your horse
You will lose your life
End

How to arrange the answers?

- Following the principle “The further to the right, the worse it is”

  or

- Following the principle “The further to the right, the larger it is”

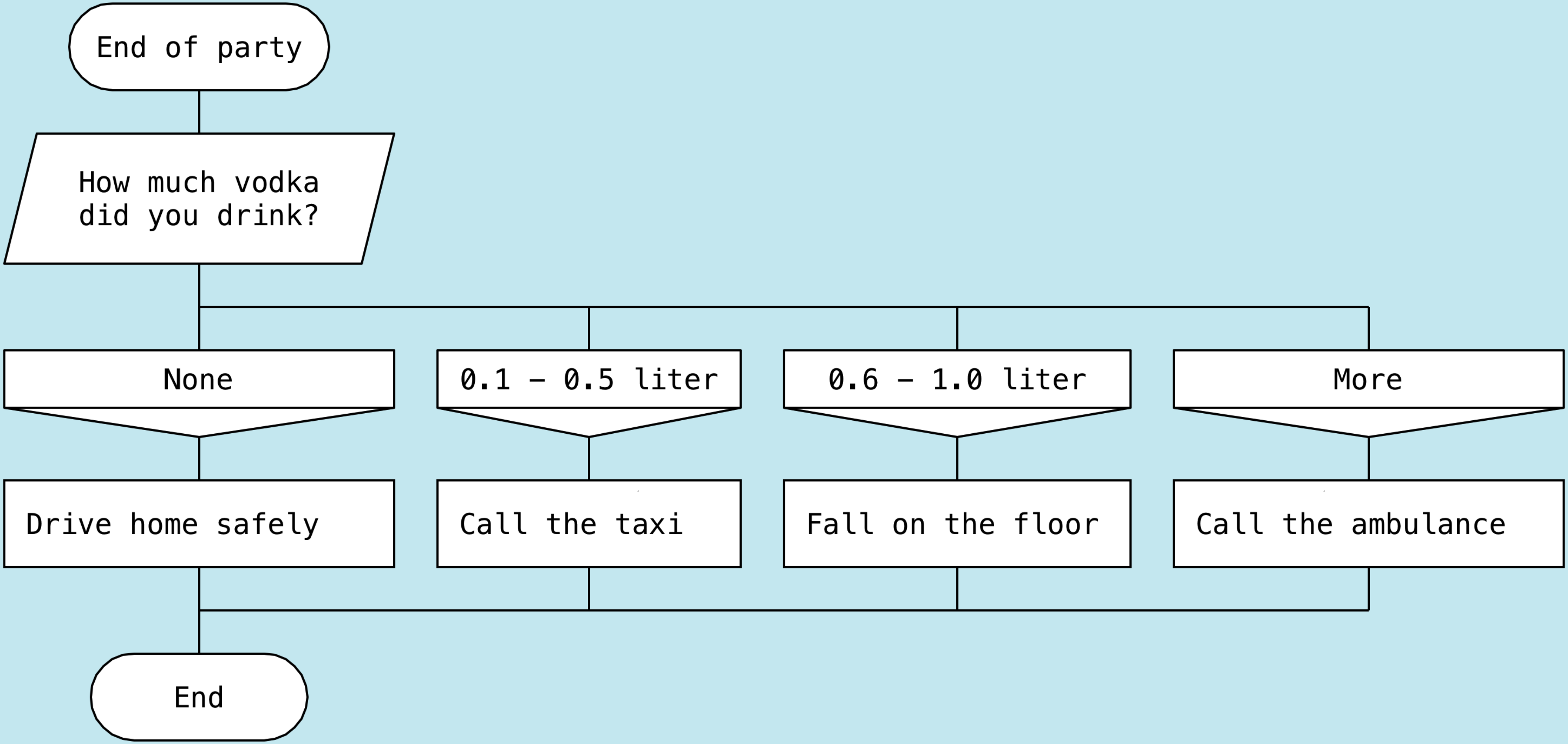
  or

- Following the principle “The further to the right, the higher it is”

  or...

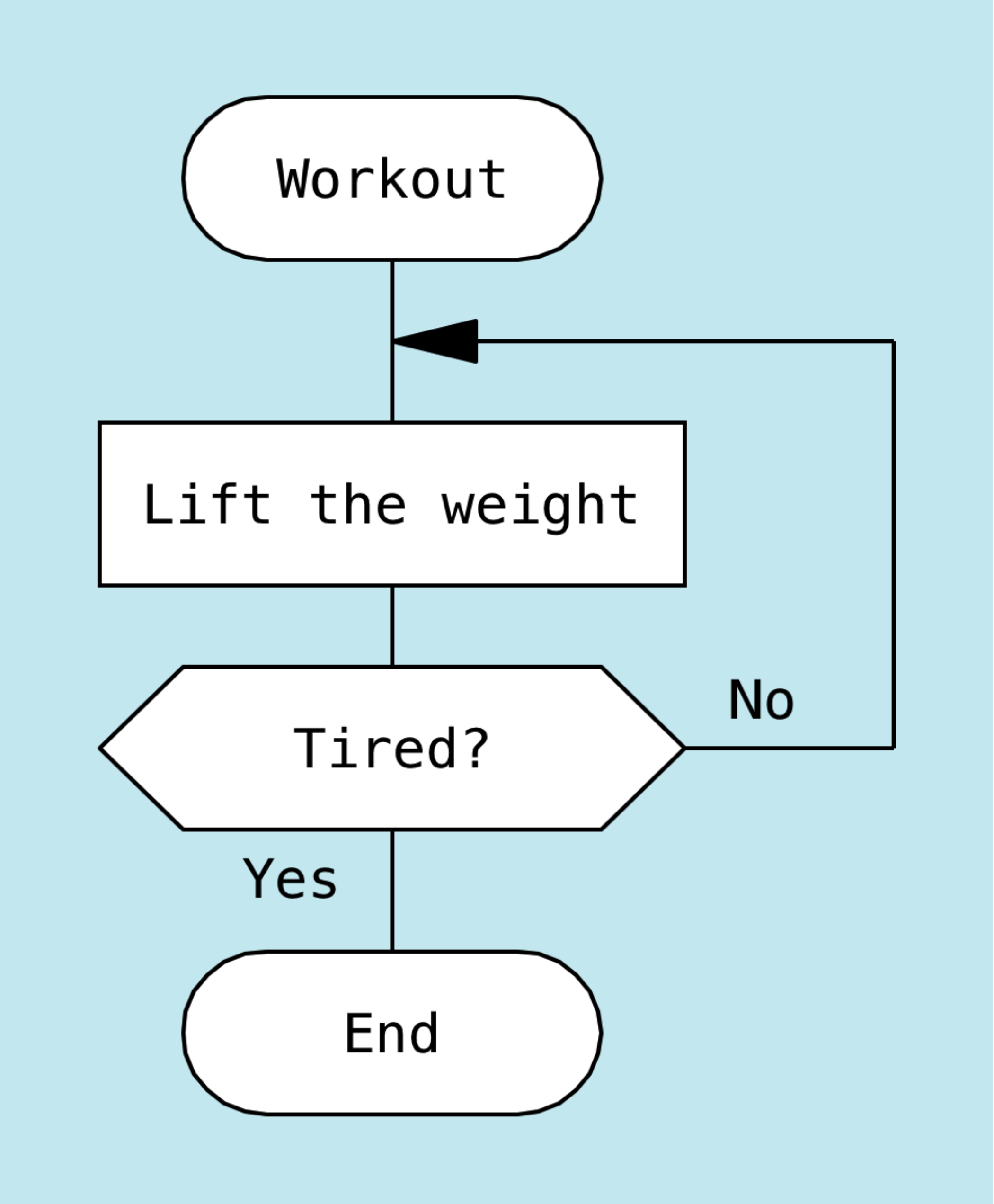
The key thing is to sort the answers

The further to the right, the larger it is

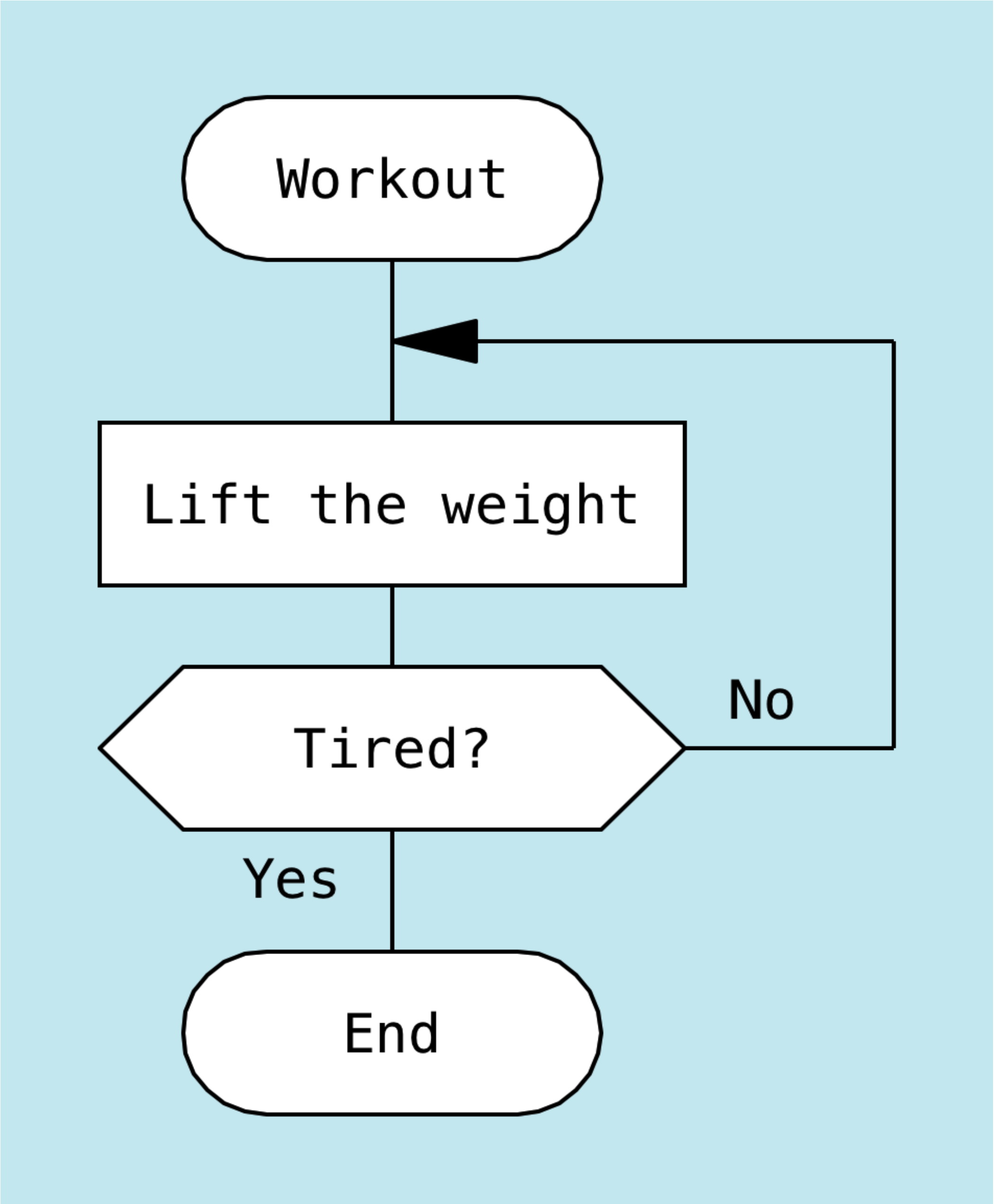


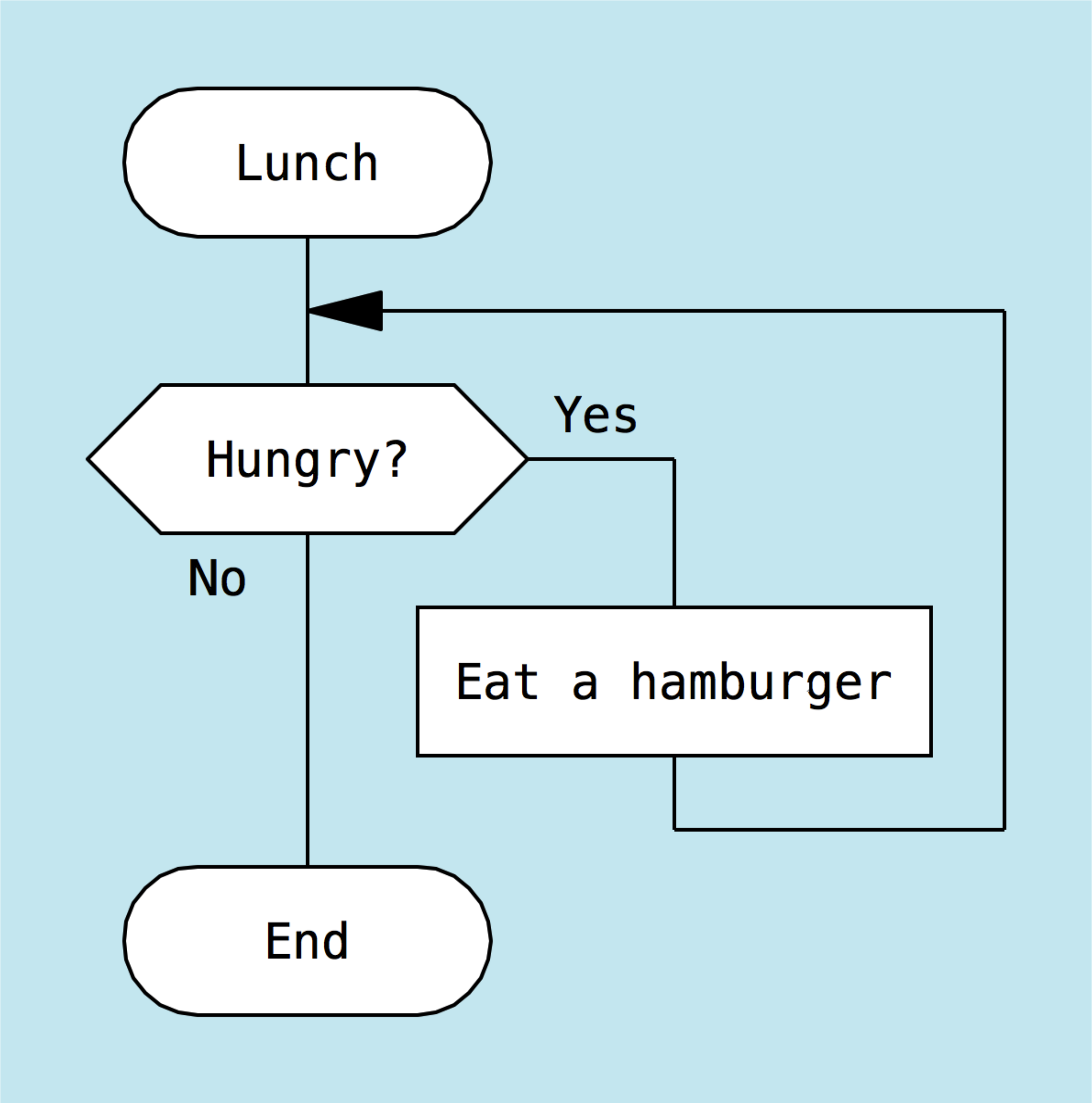
Loops

Lift the weight while not tired



The body of this loop is run at least once



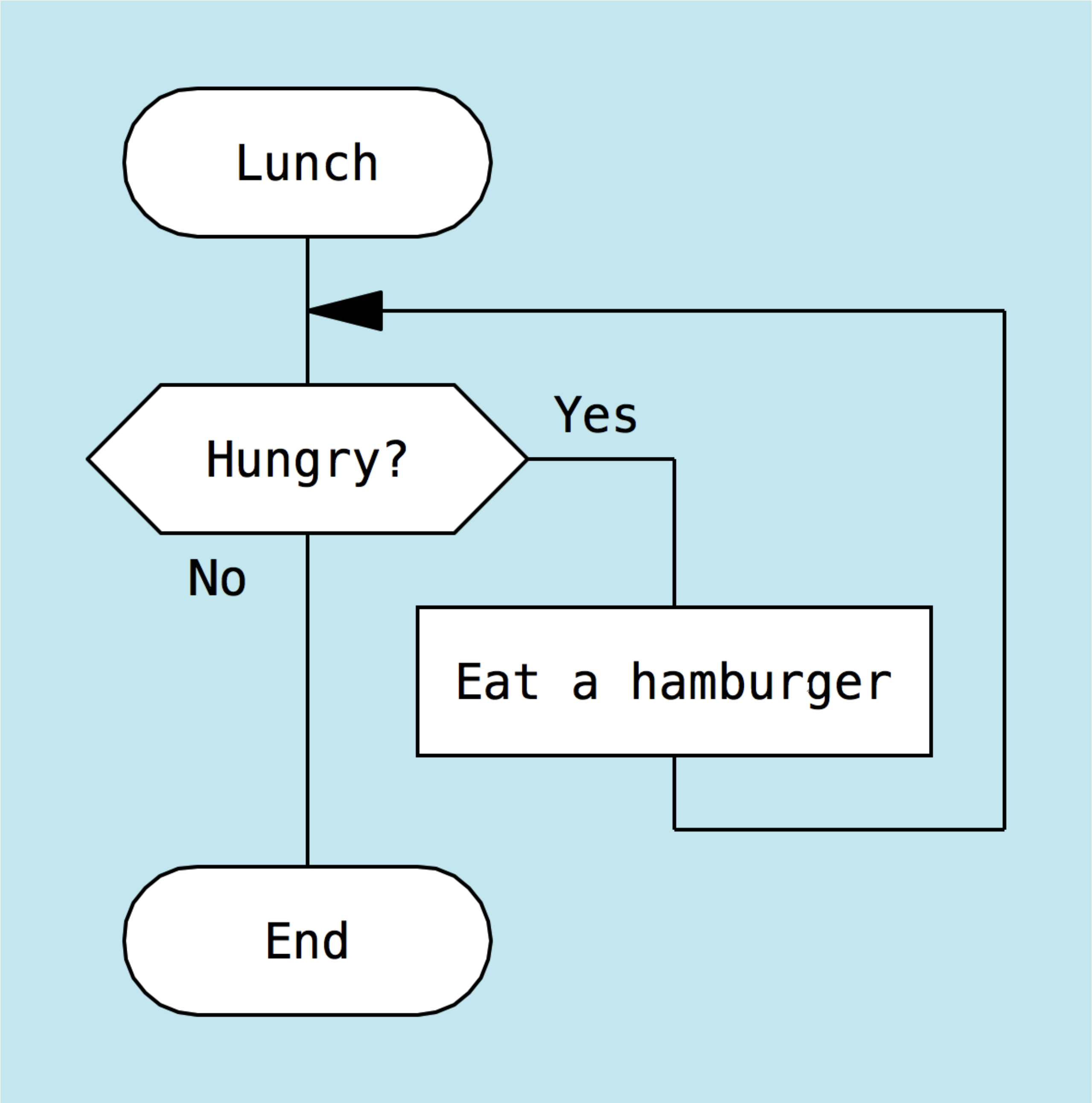


Hungry?

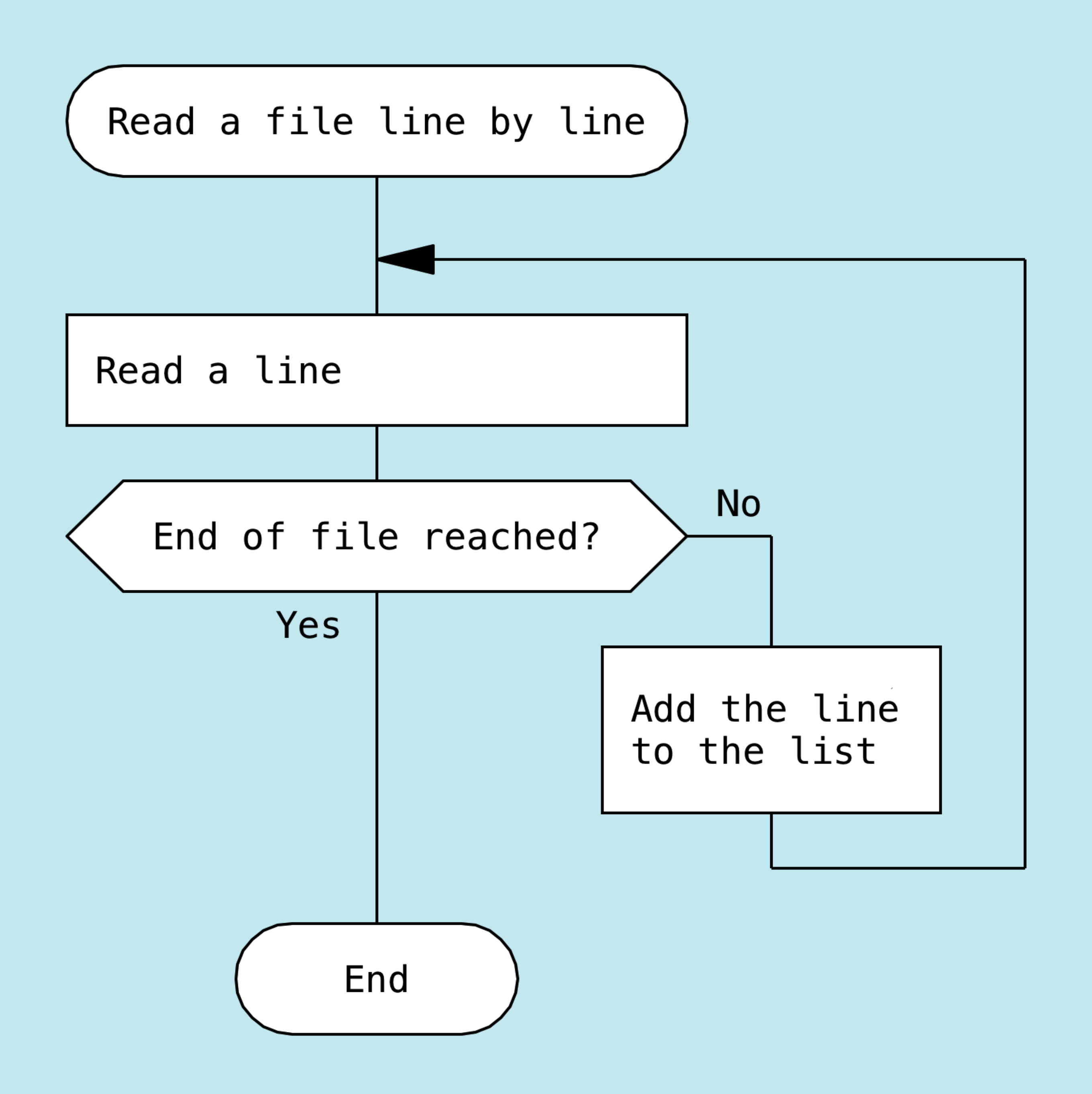
If yes, then
eat a hamburger
start from the
beginning

If no, then leave

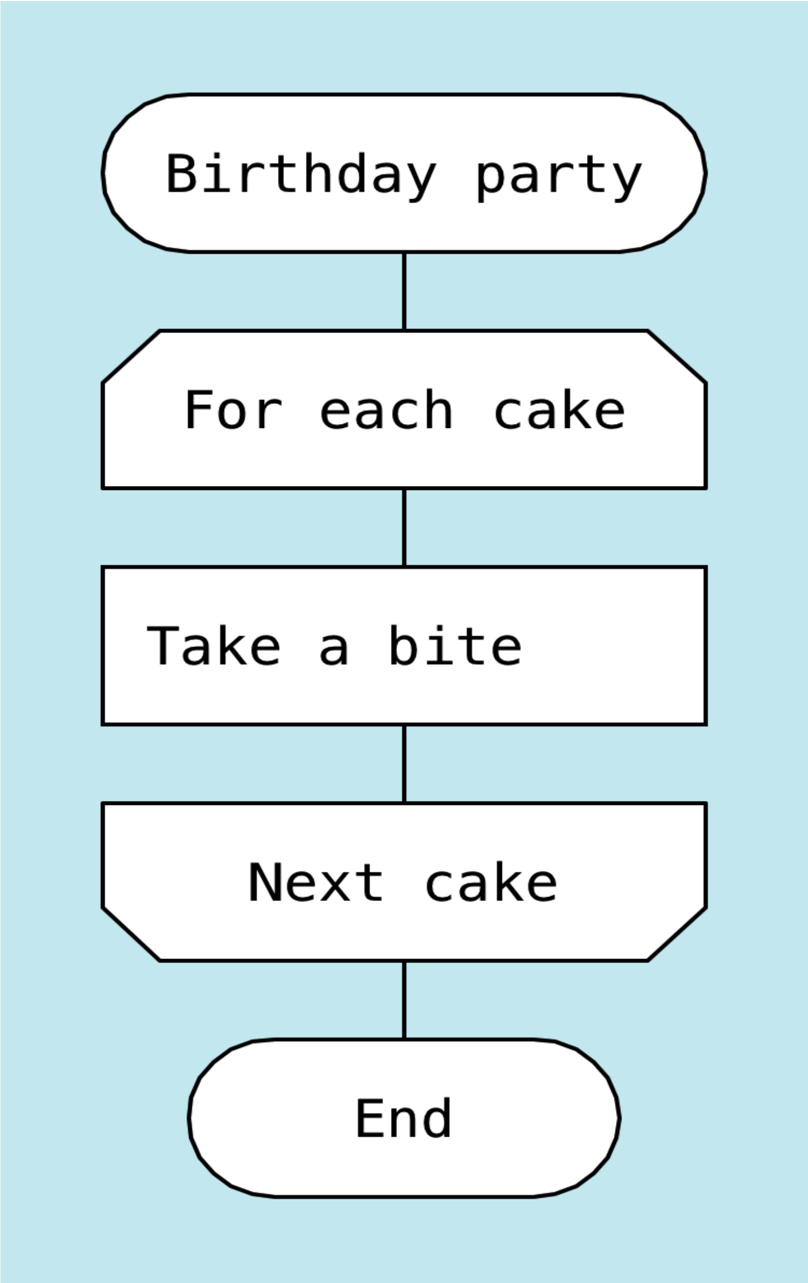
This loop is skipped if the condition is not met



The loop condition is inside the loop body



“For each” loop



Birthday party

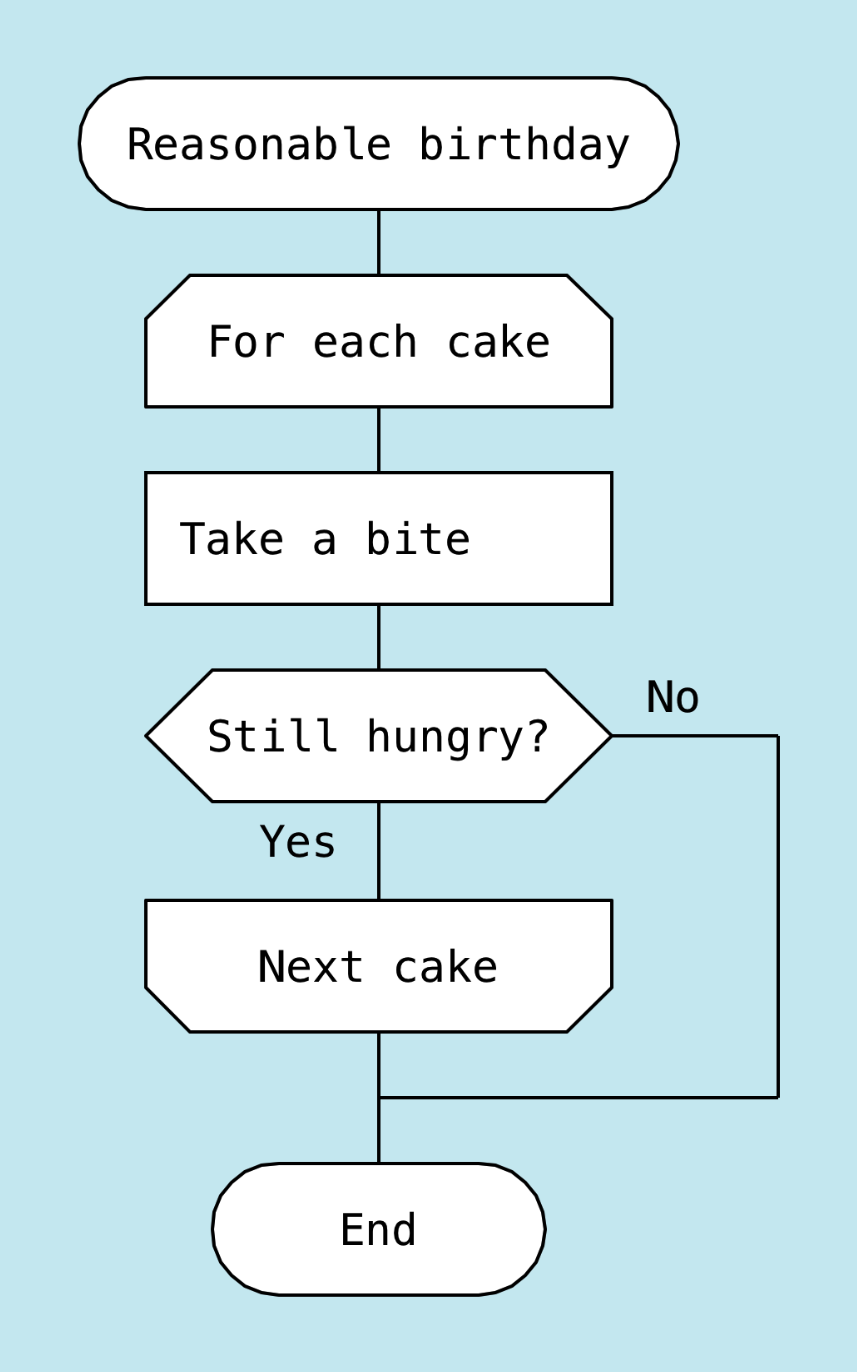
For each cake

Take a bite

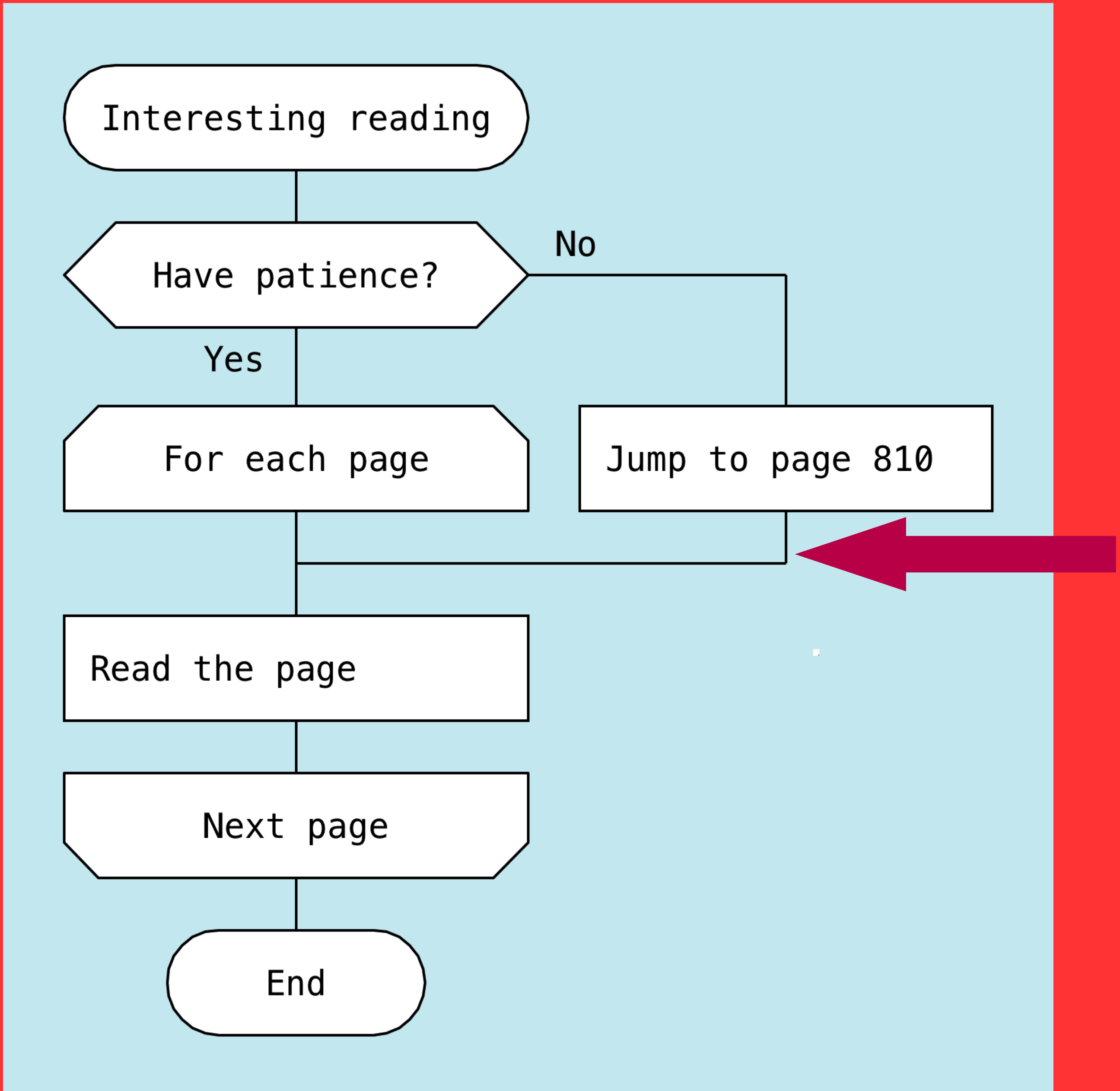
Next cake

End

There can be several exits from a loop

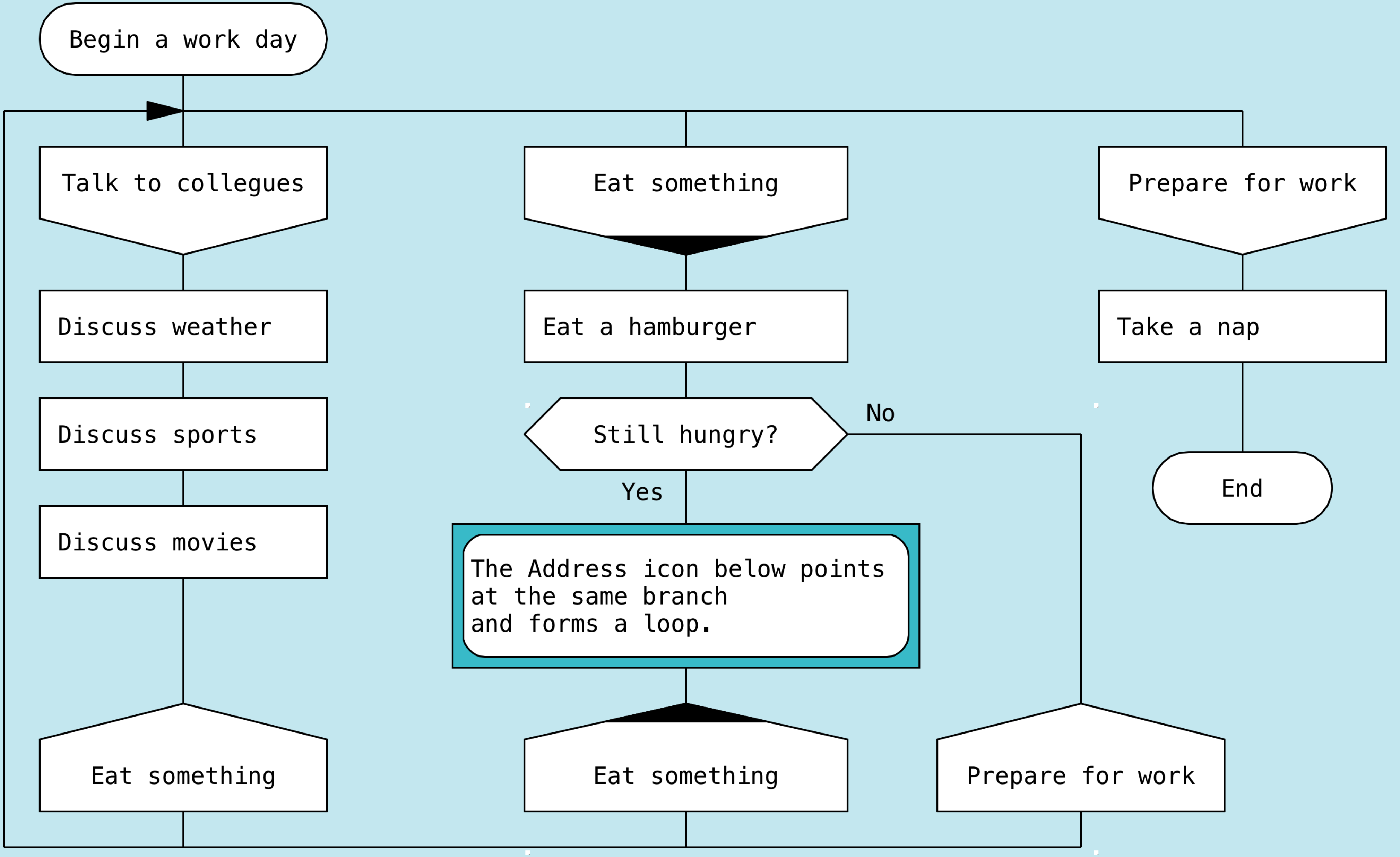


There can be only ONE ENTRANCE into a loop



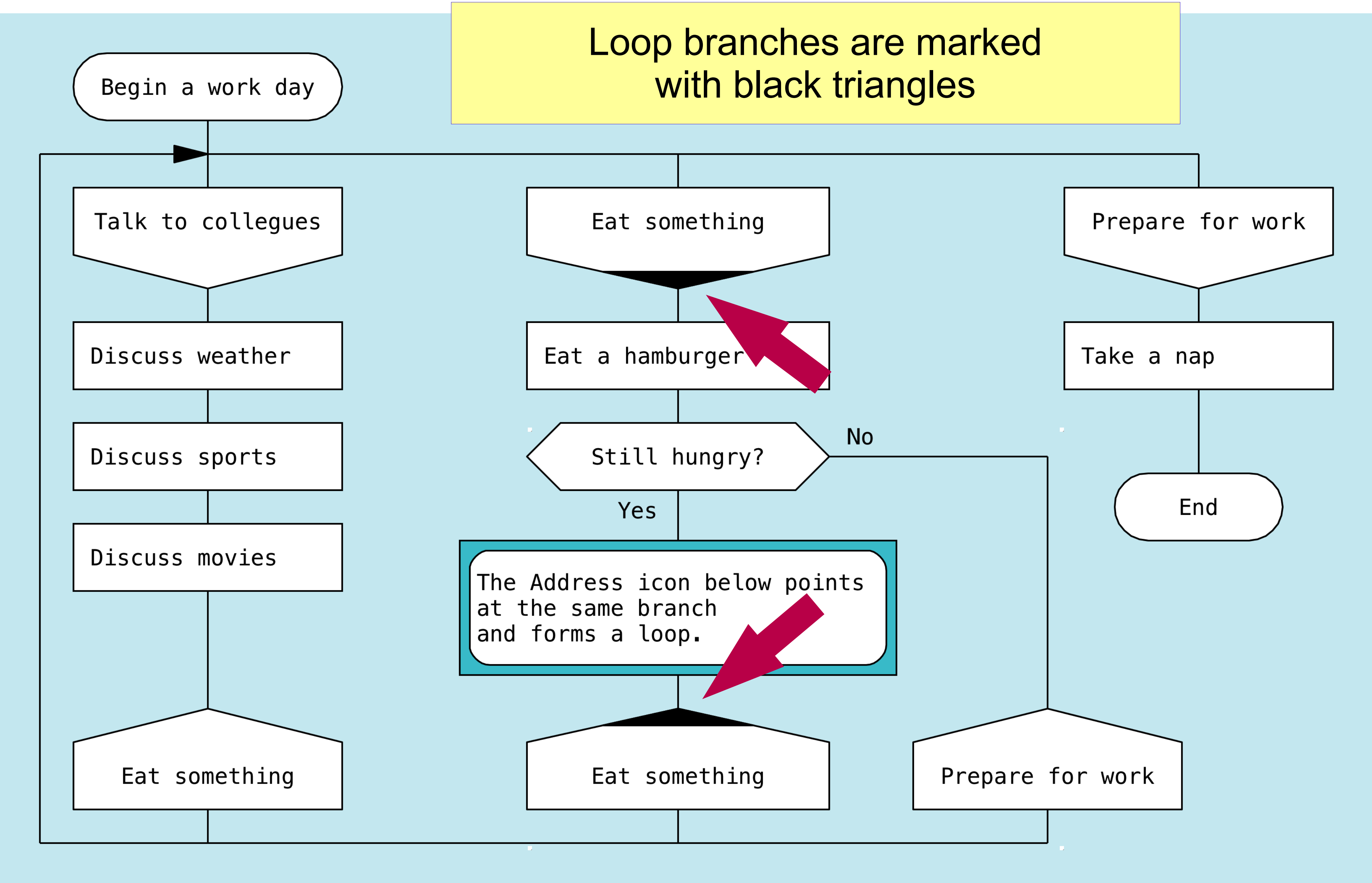
Wrong!
A second entrance is not allowed

Silhouette loop: some branches are run several times



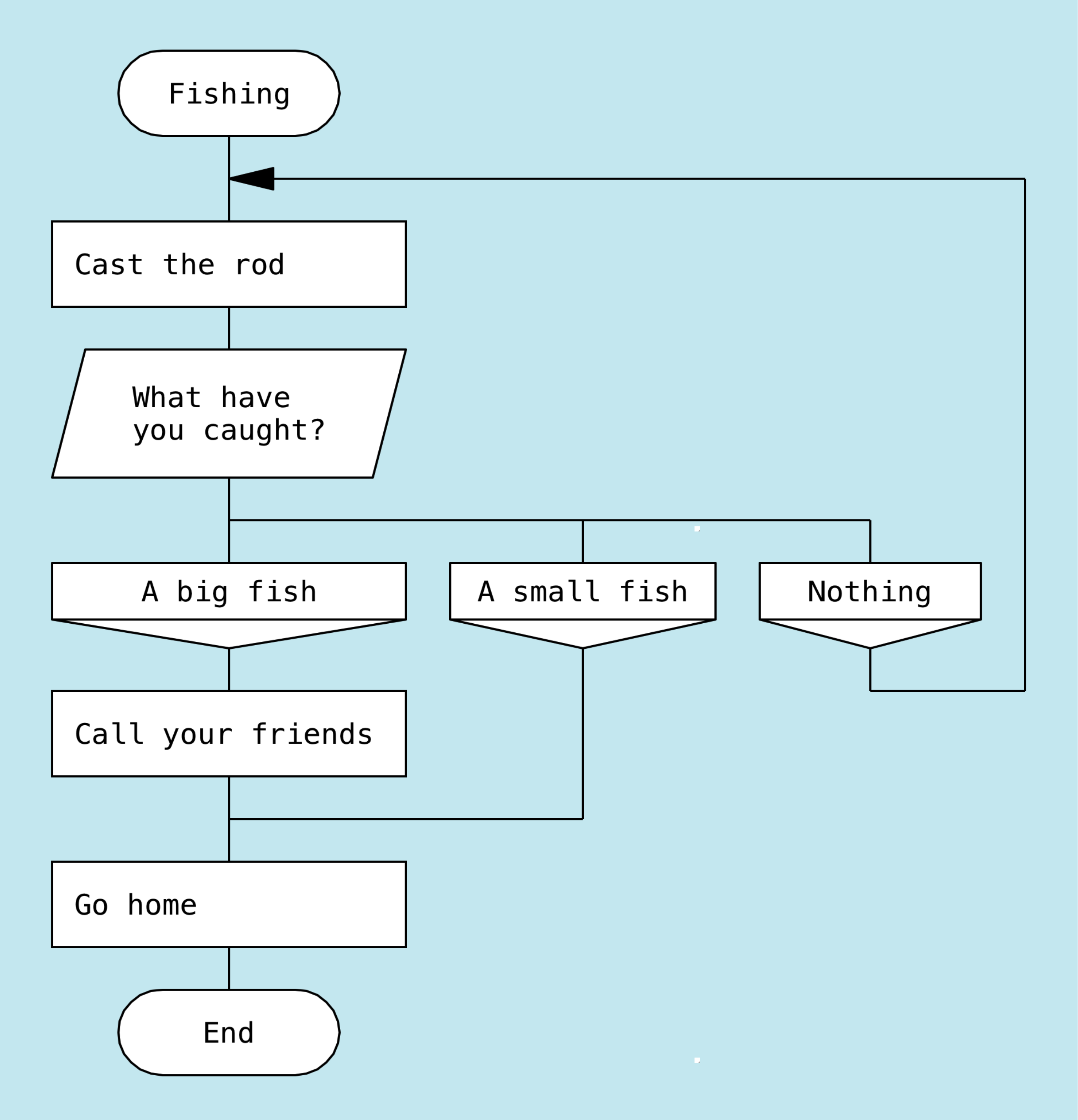
Silhouette loop: some branches are run several times

Loop branches are marked
with black triangles



The Address icon below points
at the same branch
and forms a loop.

“Select” loop



Fishing

Cast the rod

What have you caught?

A big fish

A small fish

Nothing

Call your friends

Go home

End

End

Stepan Mitkin

stipan.mitkin@gmail.com

The diagrams on the slides were made with DRAKON Editor

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

June 2013