

What is static analysis?

Matt Might

matt.might.net

University of Utah

@mattmight



interpreter

abstract interpreter

static analyzer

What

why

Too slow.

Too buggy.

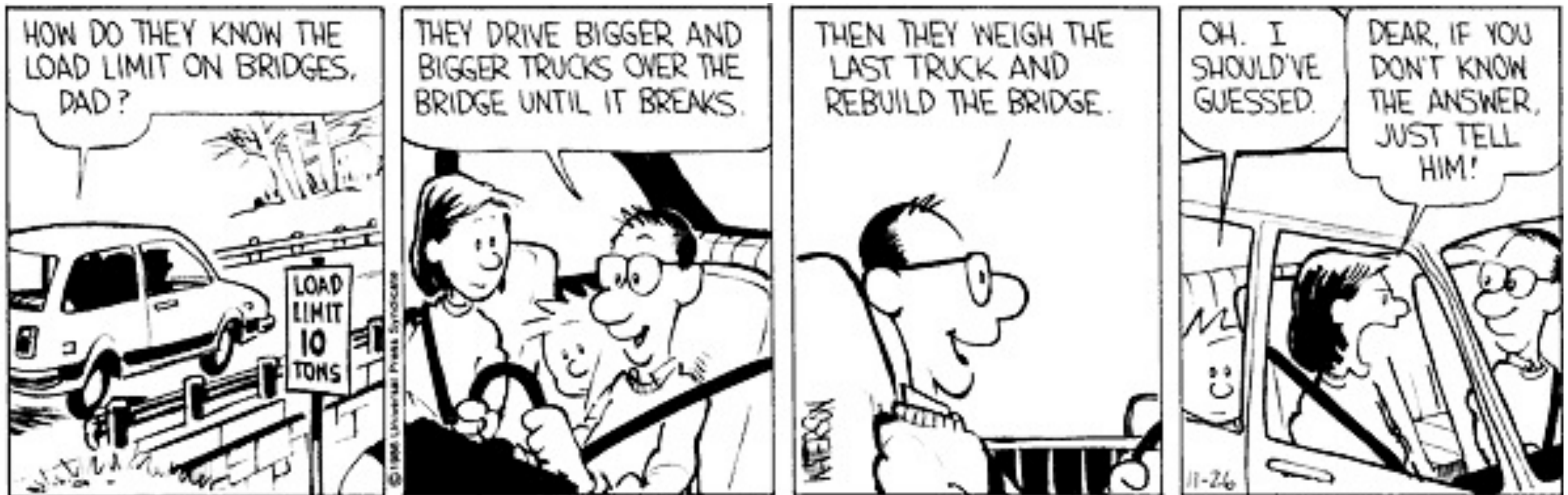
Too insecure.

Why?

We can't engineer.

Software engineering?

Software engineering?



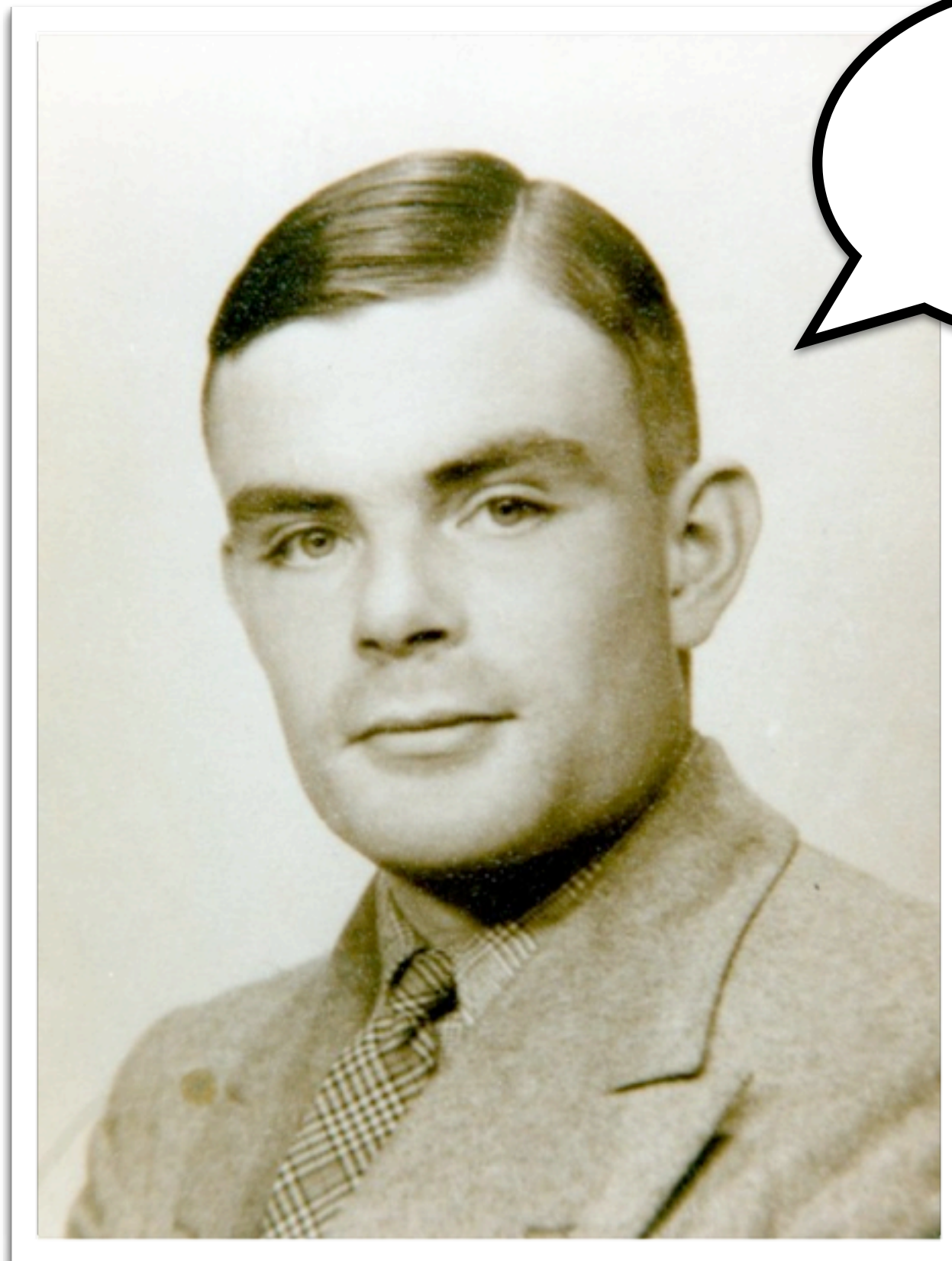
We can't engineer.

We can't predict.

Why?

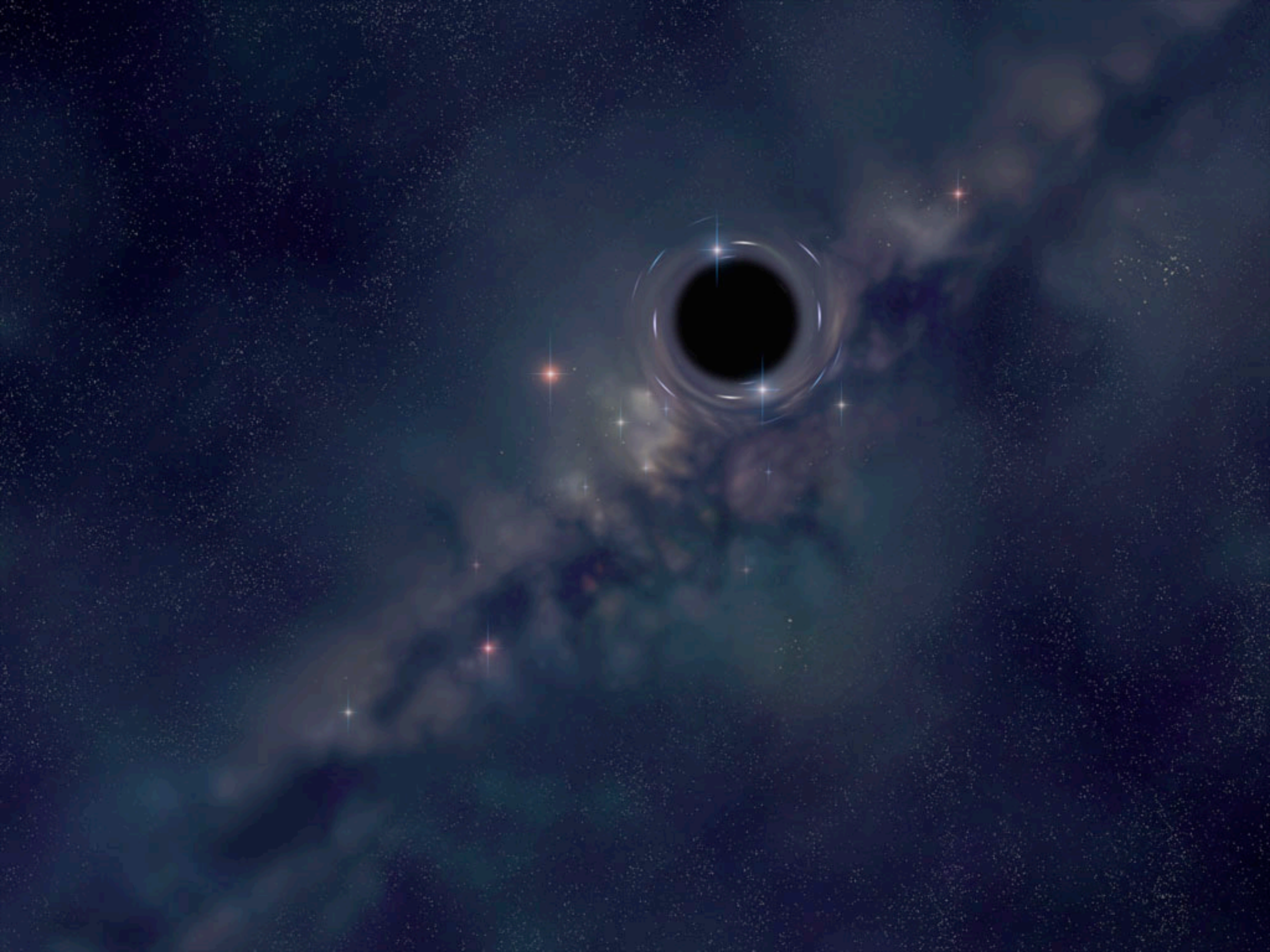


Because Alan Turing said so!



Halt!

“Thou shalt not write an algorithm which determines whether a program halts.”



“the loop hole”



while $P(x)$

Interesting question?

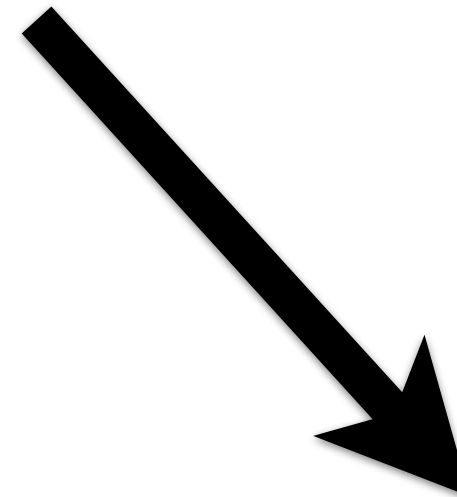
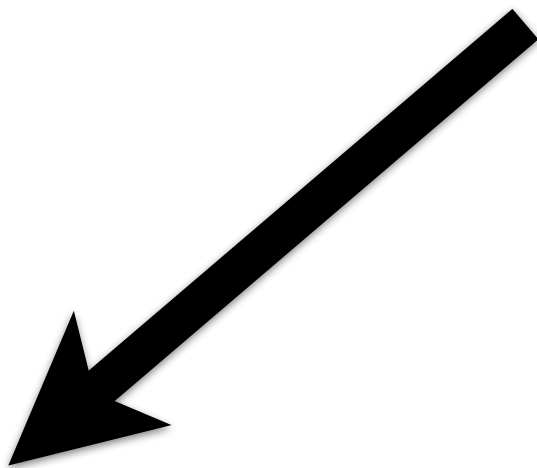
Interesting question?

Undecidable.

Why we need
software *engineering*

```
class MyActivity {  
  
    public MyActivity() {  
        activateMic();  
    }  
  
}
```

```
class MyActivity {  
  
    public MyActivity() {  
        activateMic();  
    }  
  
}
```



A problem has been detected and Windows has been shut down to prevent damage to your computer.

The problem seems to be caused by the following file: SPOMDCON.SYS

PAGE_FAULT_IN_NONPAGED_AREA

If this is the first time you've seen this stop error screen, restart your computer. If this screen appears again, follow these steps:

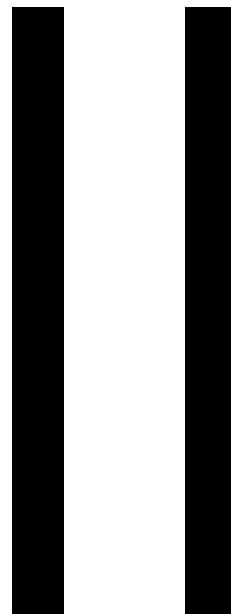
Check to make sure any new hardware or software is properly installed. If this is a new installation, ask your hardware or software manufacturer for any Windows updates you might need.

If problems continue, disable or remove any newly installed hardware or software. Disable BIOS memory options such as caching or shadowing. If you need to use Safe Mode to remove or disable components, restart your computer, press F8 to select Advanced Startup options, and then select Safe Mode.

Technical information:

*** STOP: 0x00000050 (0xFD3094C2, 0x00000001, 0xFBFE7617, 0x00000000)

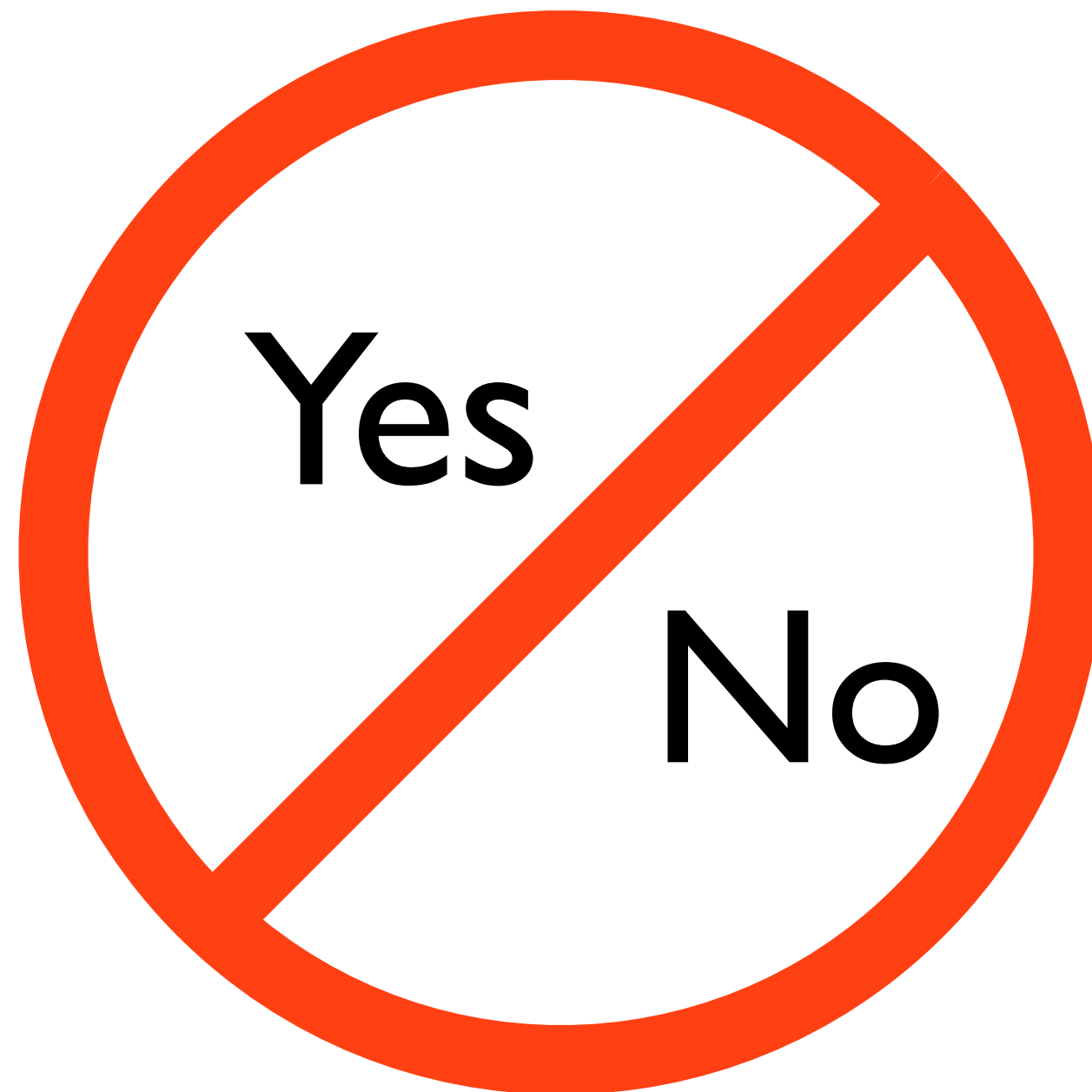
*** SPOMDCON.SYS - Address FBFE7617 base at FBFE5000, DateStamp 3d6dd67c

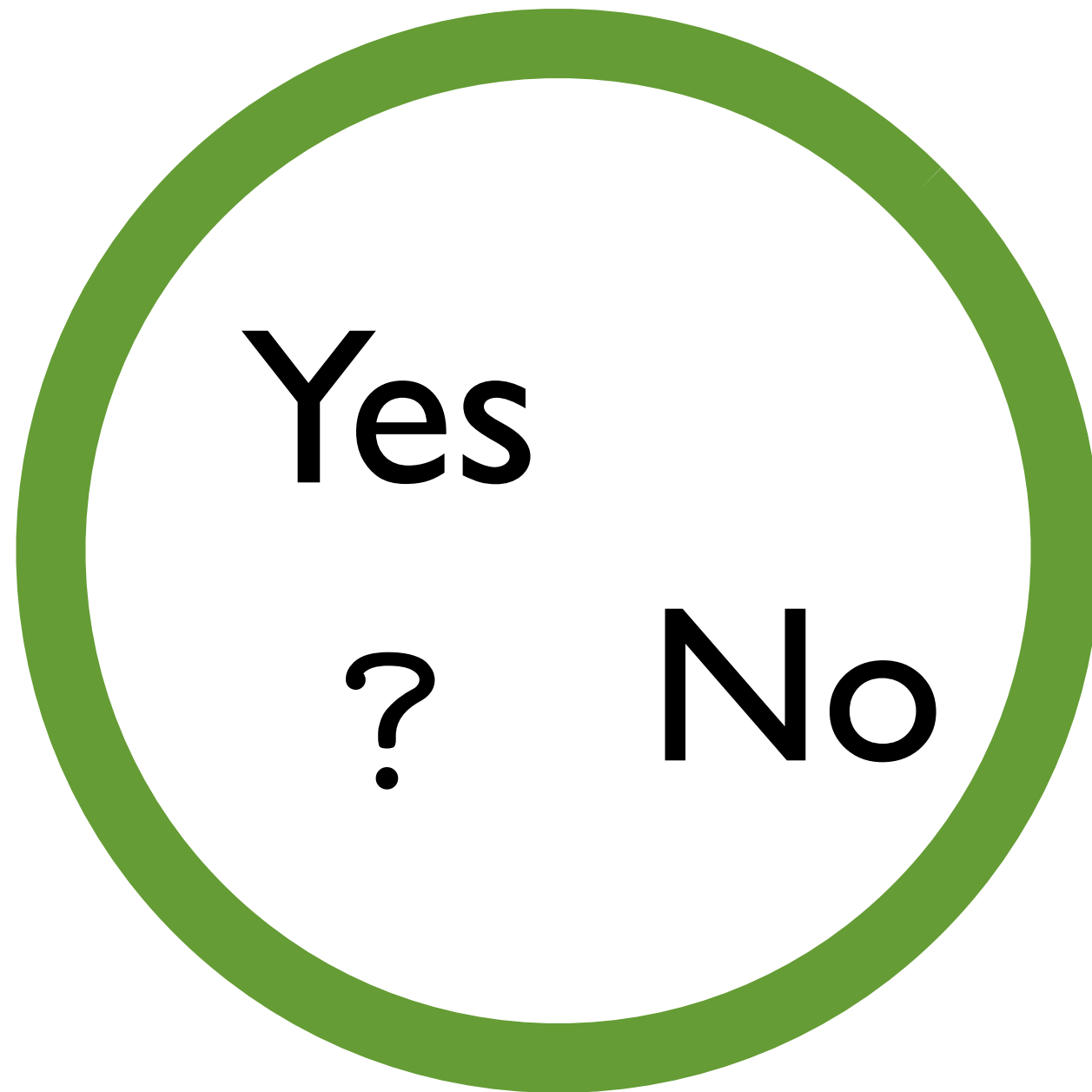


There's a loop hole...

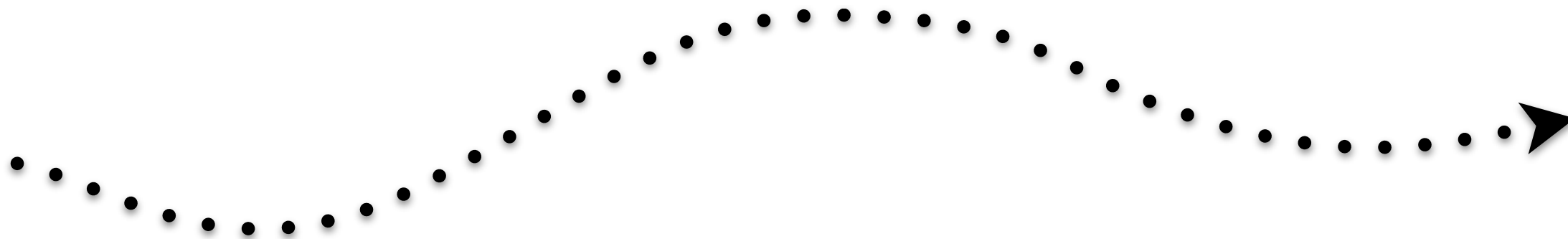
There's a loop hole...

...in the loop hole.





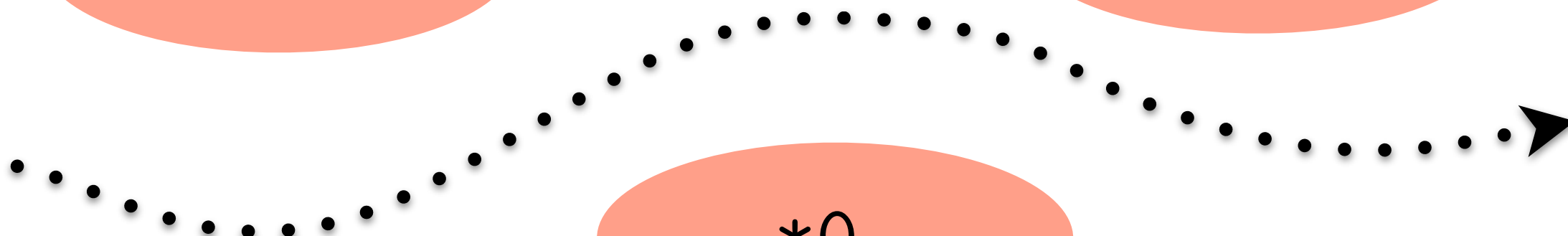
static analysis = reasoning

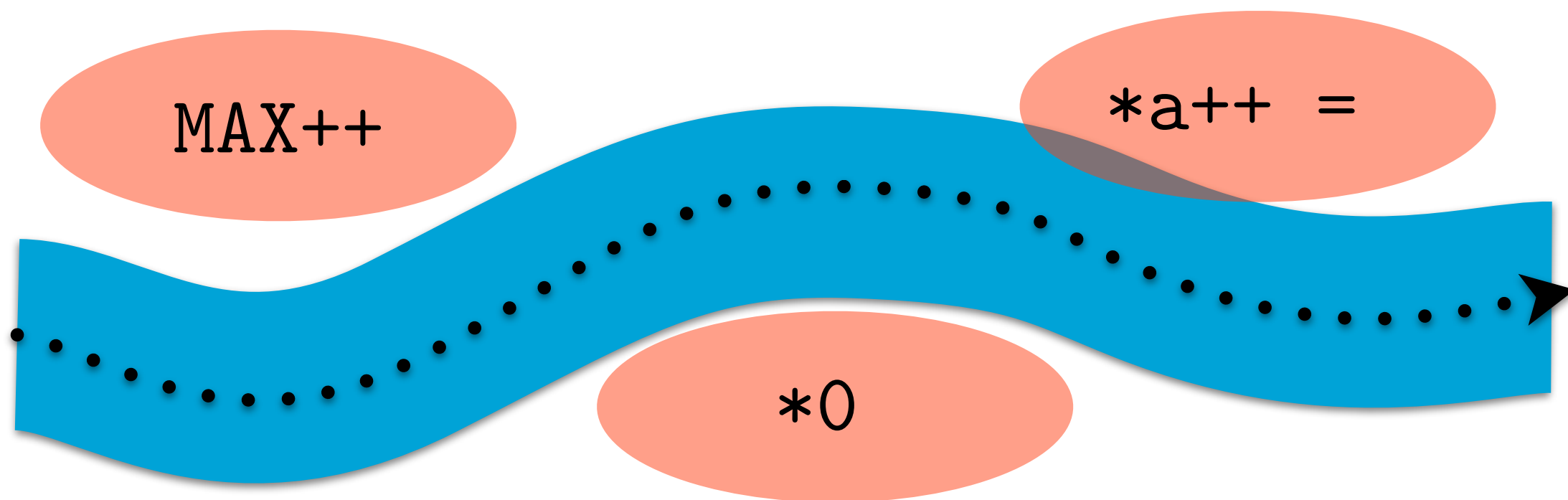


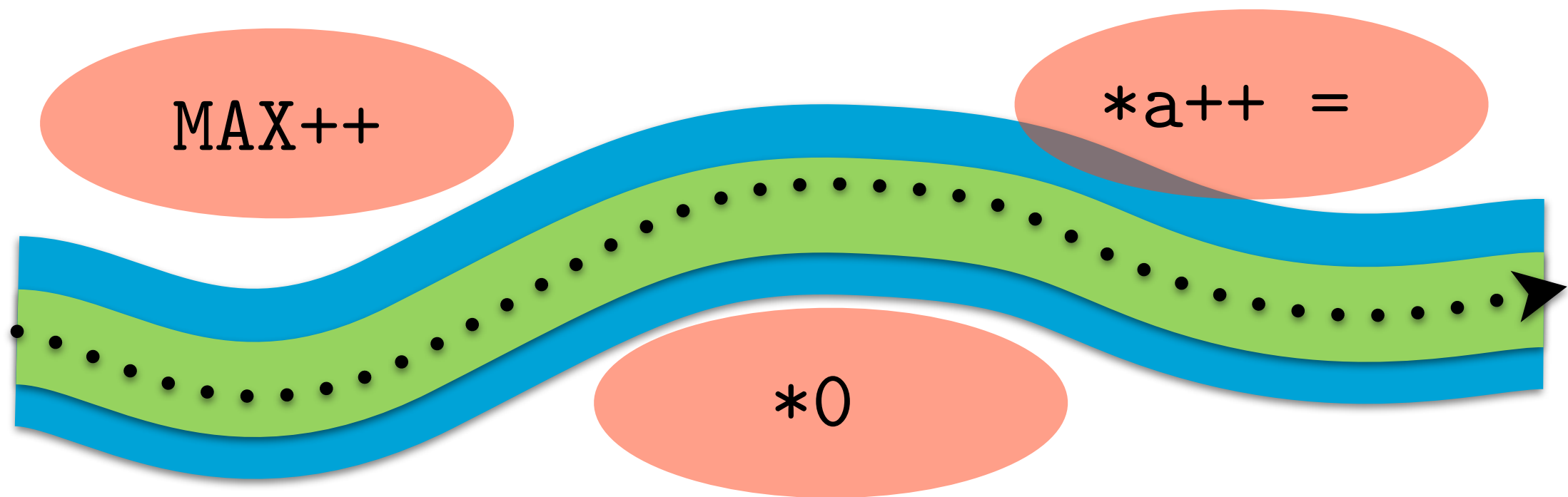
MAX++

*a++ =

*0





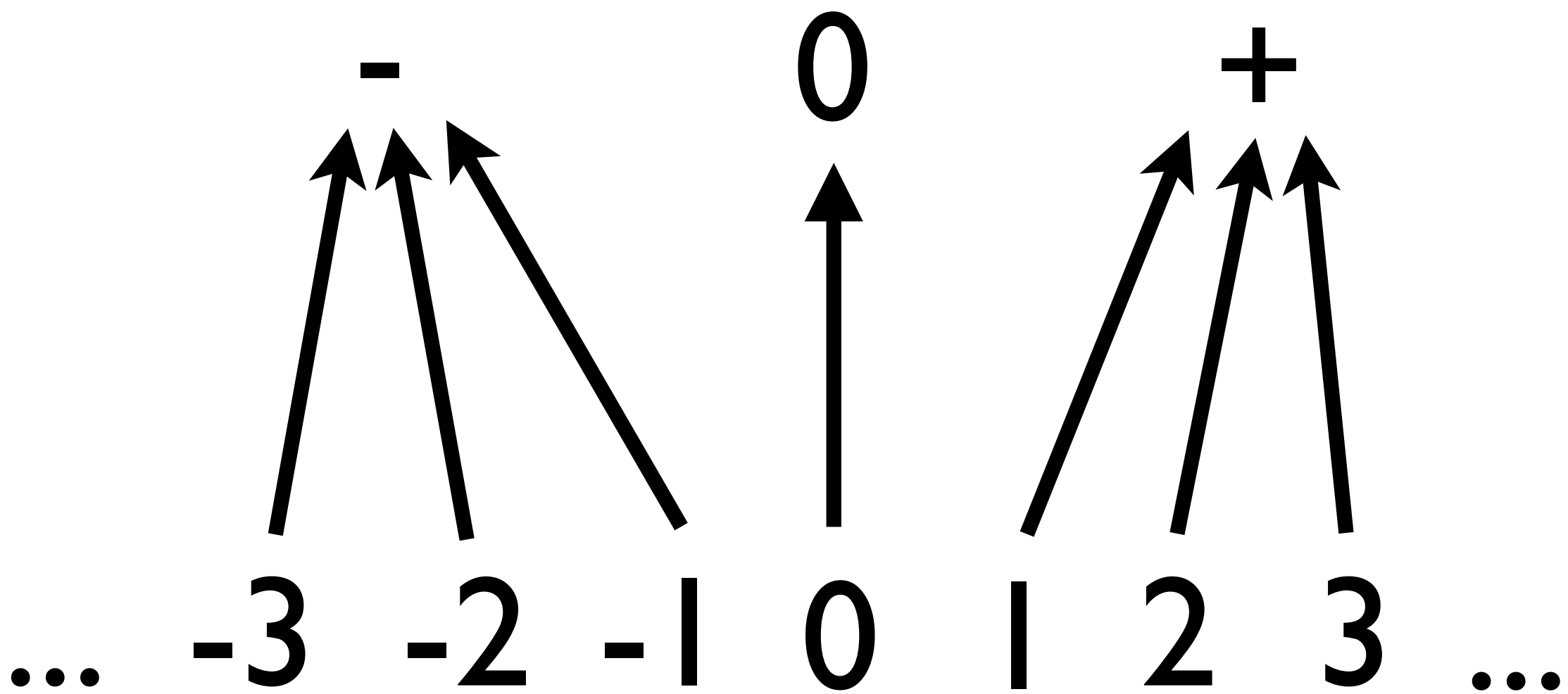


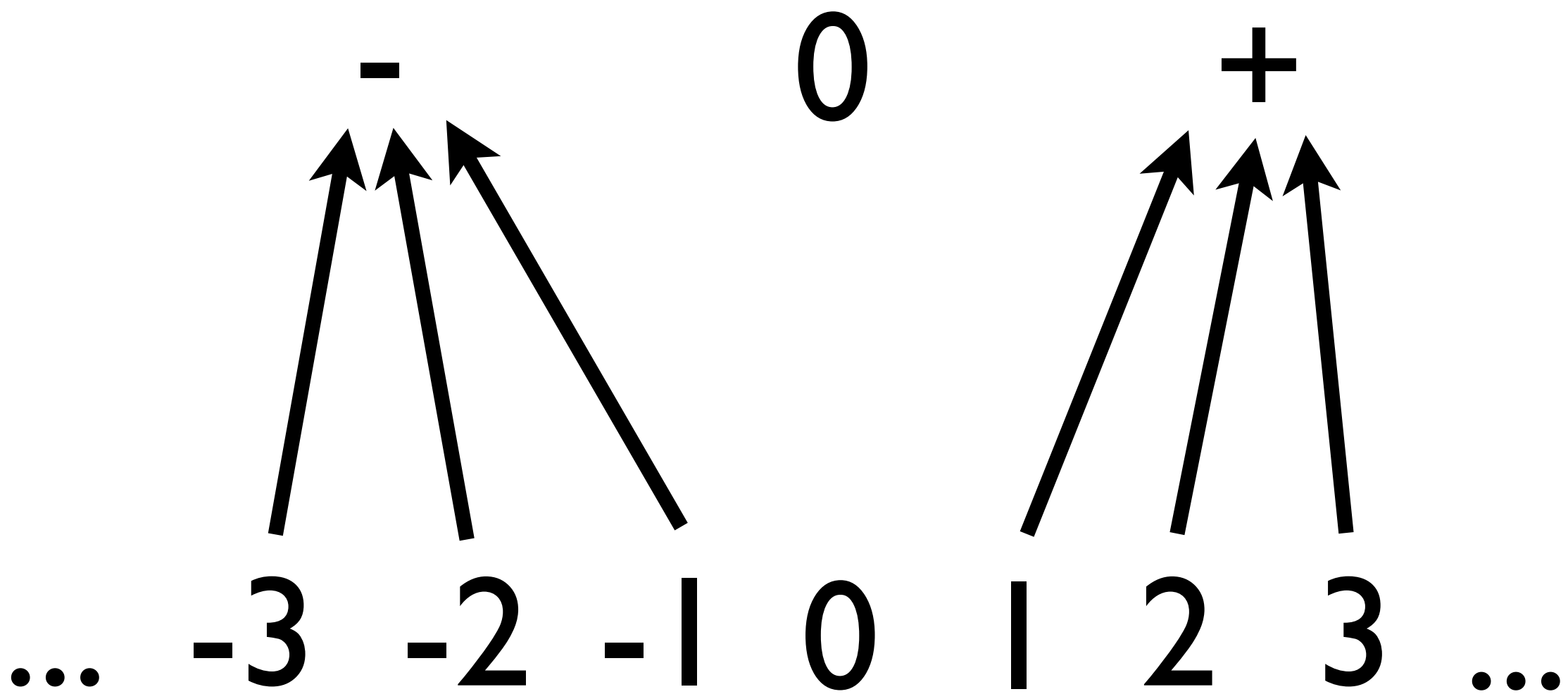
Example: Sign analysis

What is the sign of -3×2 ?

... -3 -2 -1 0 1 2 3 ...

... -3 -2 -1 0 1 2 3 ...





-	x	+
↑		↑
-3	x	2

$$\begin{array}{ccccccccc} & - & & \times & & + & & = & & - \\ & \uparrow & & & & \uparrow & & & & \uparrow \\ -3 & & & \times & & 2 & & = & & -6 \end{array}$$

Let's build it!

$\langle \text{exp} \rangle ::= \langle \text{int} \rangle$
 $| \langle \text{exp} \rangle * \langle \text{exp} \rangle$
 $| \langle \text{exp} \rangle = \langle \text{exp} \rangle$
 $| \langle \text{exp} \rangle + \langle \text{exp} \rangle$

`simple-eval : exp -> integer`

`simple-eval^ : exp -> abstract-integer`

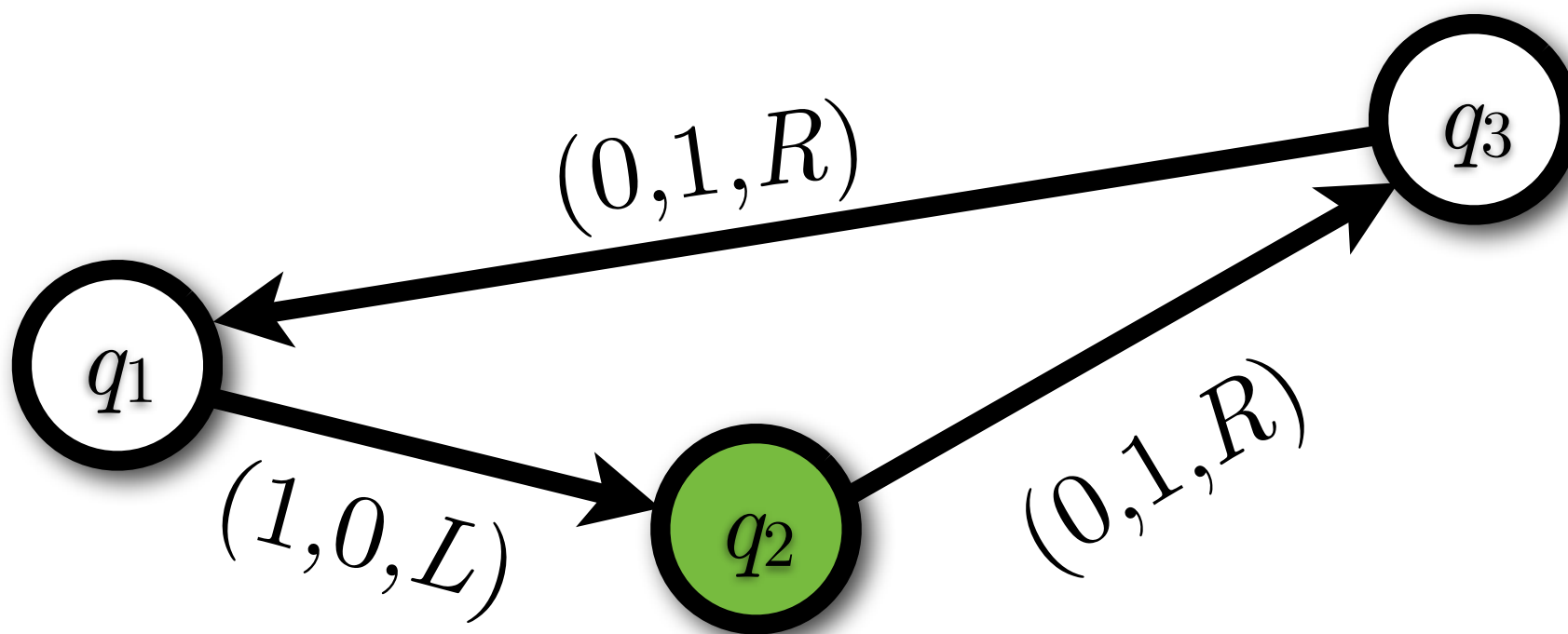
`simple-eval : exp -> integer`

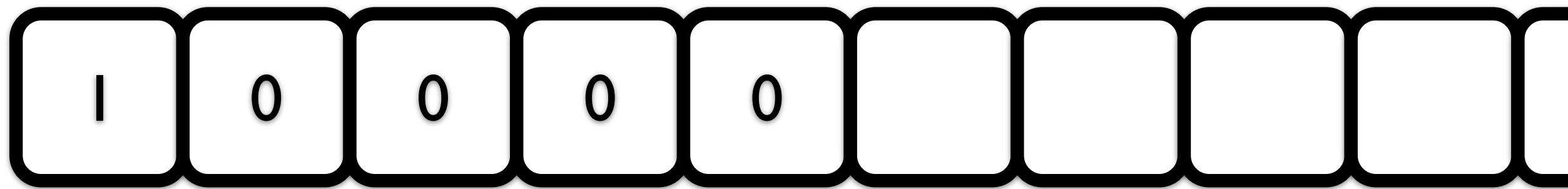
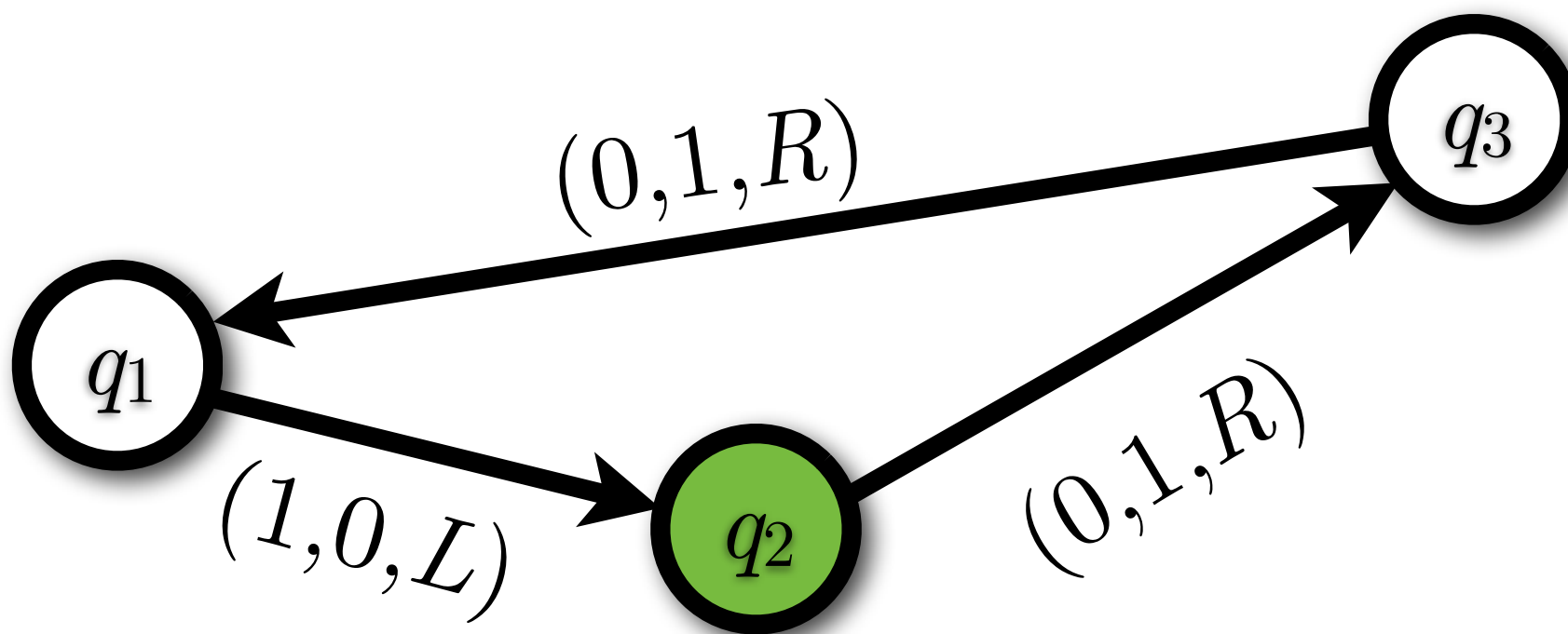
α : integer \rightarrow abstract-integer

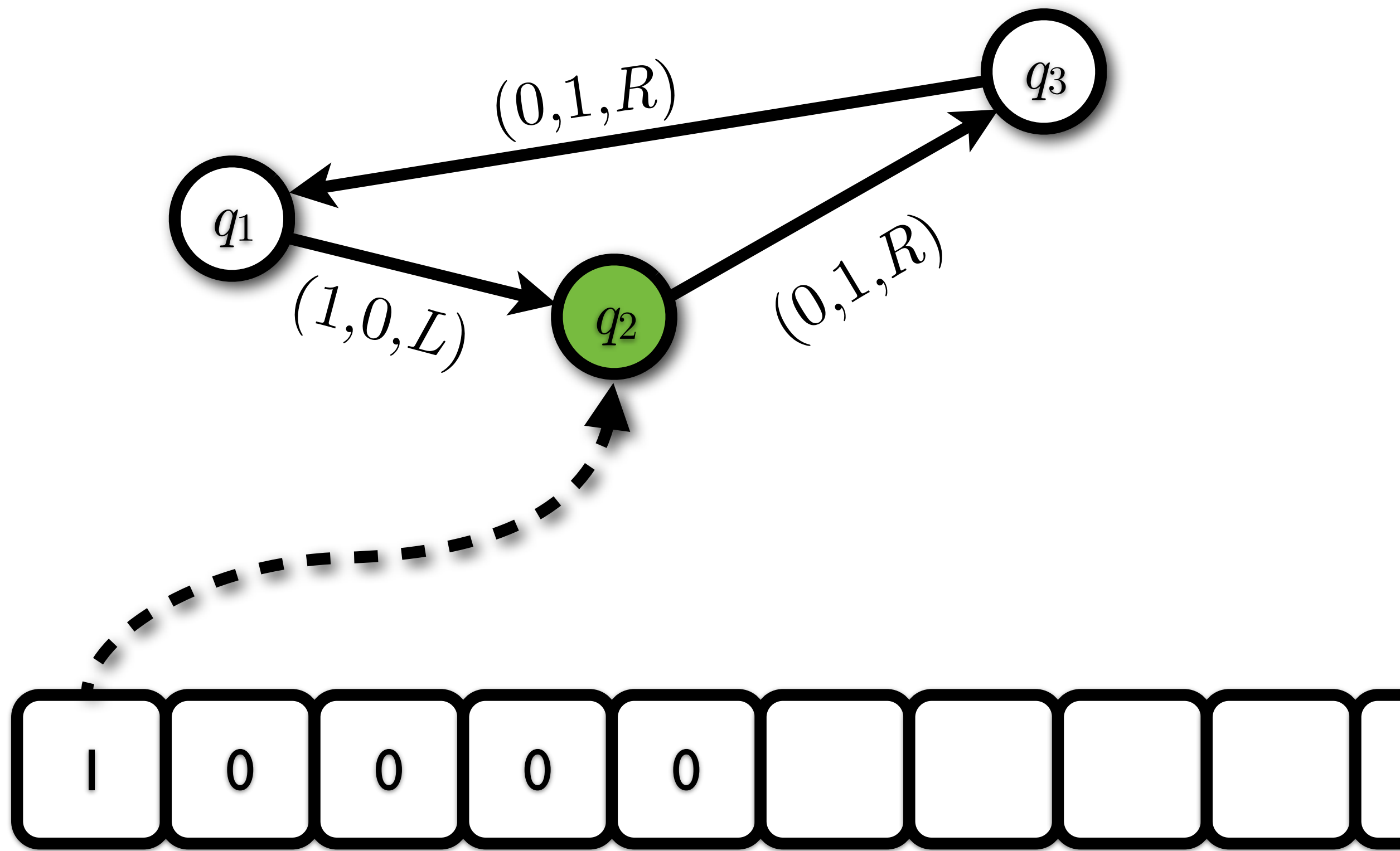
`simple-eval^ : exp -> abstract-integer`

Example: Turing machines



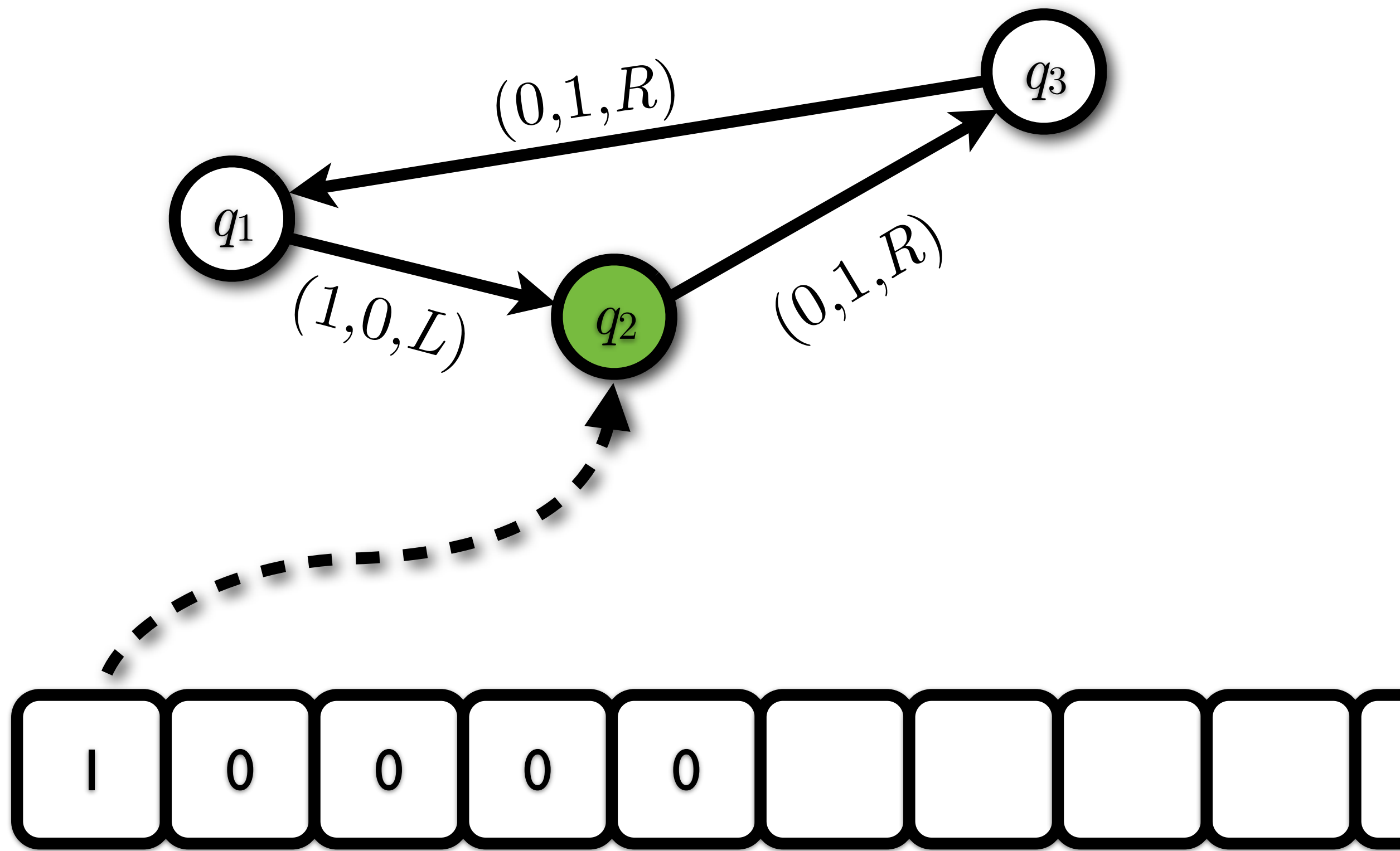


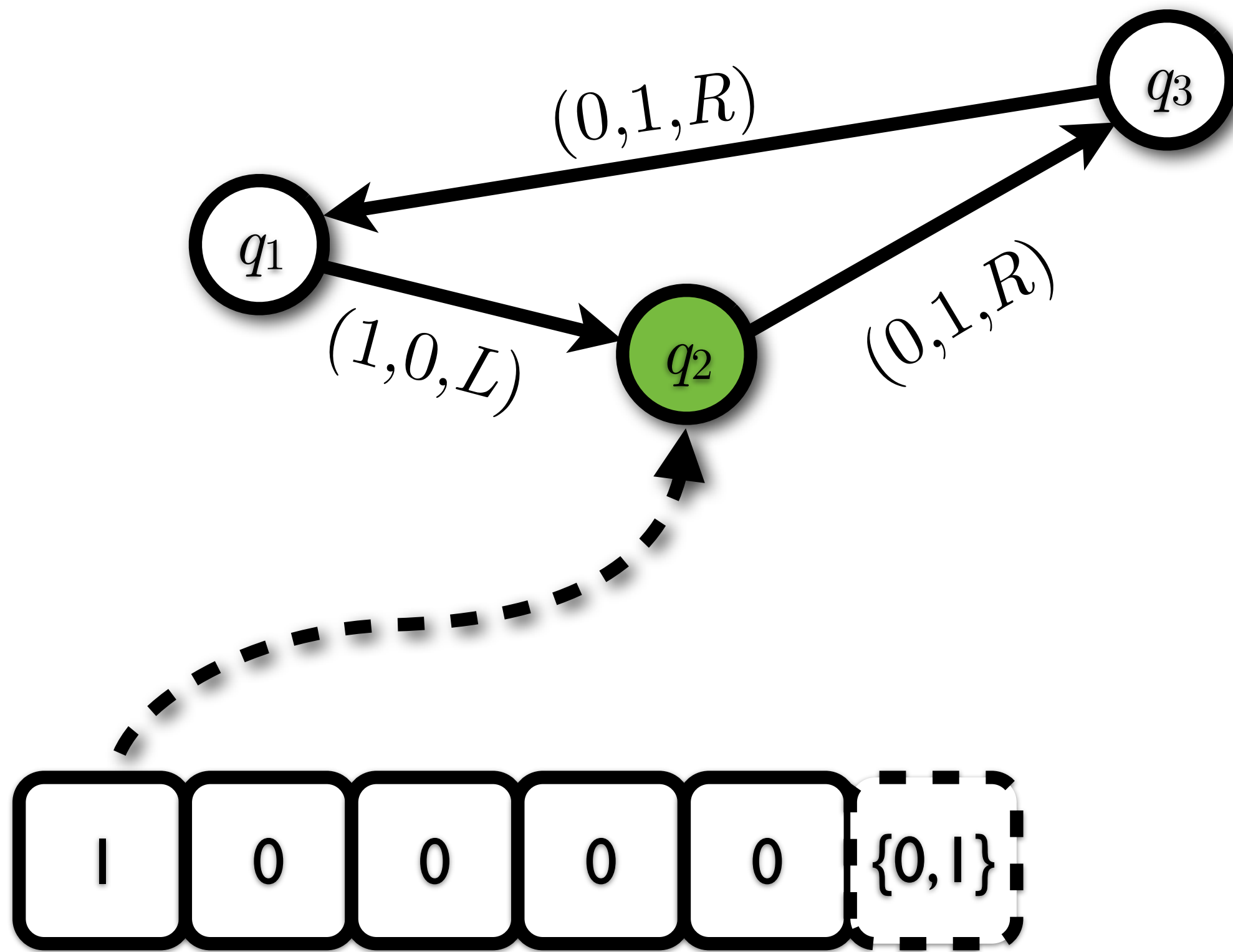


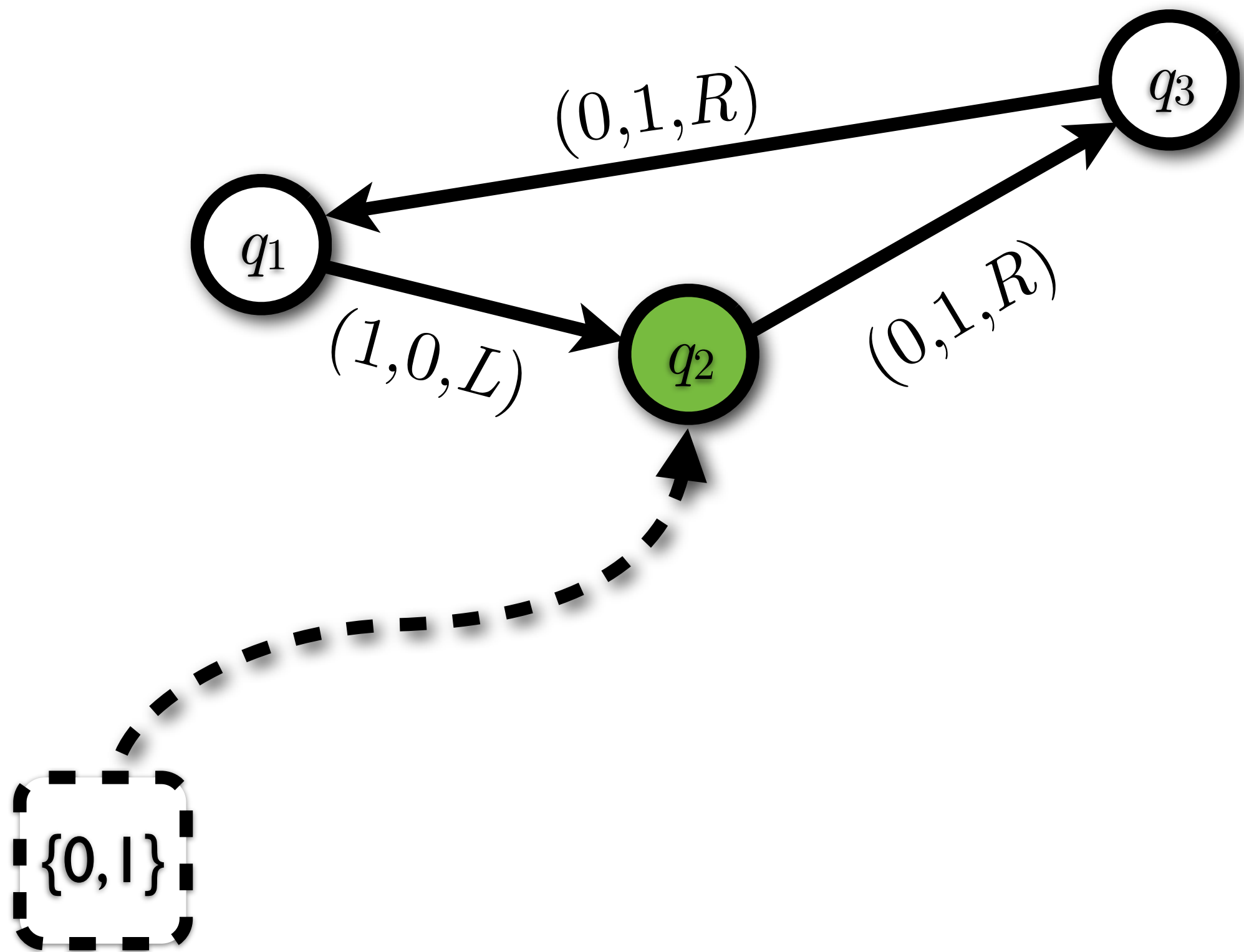


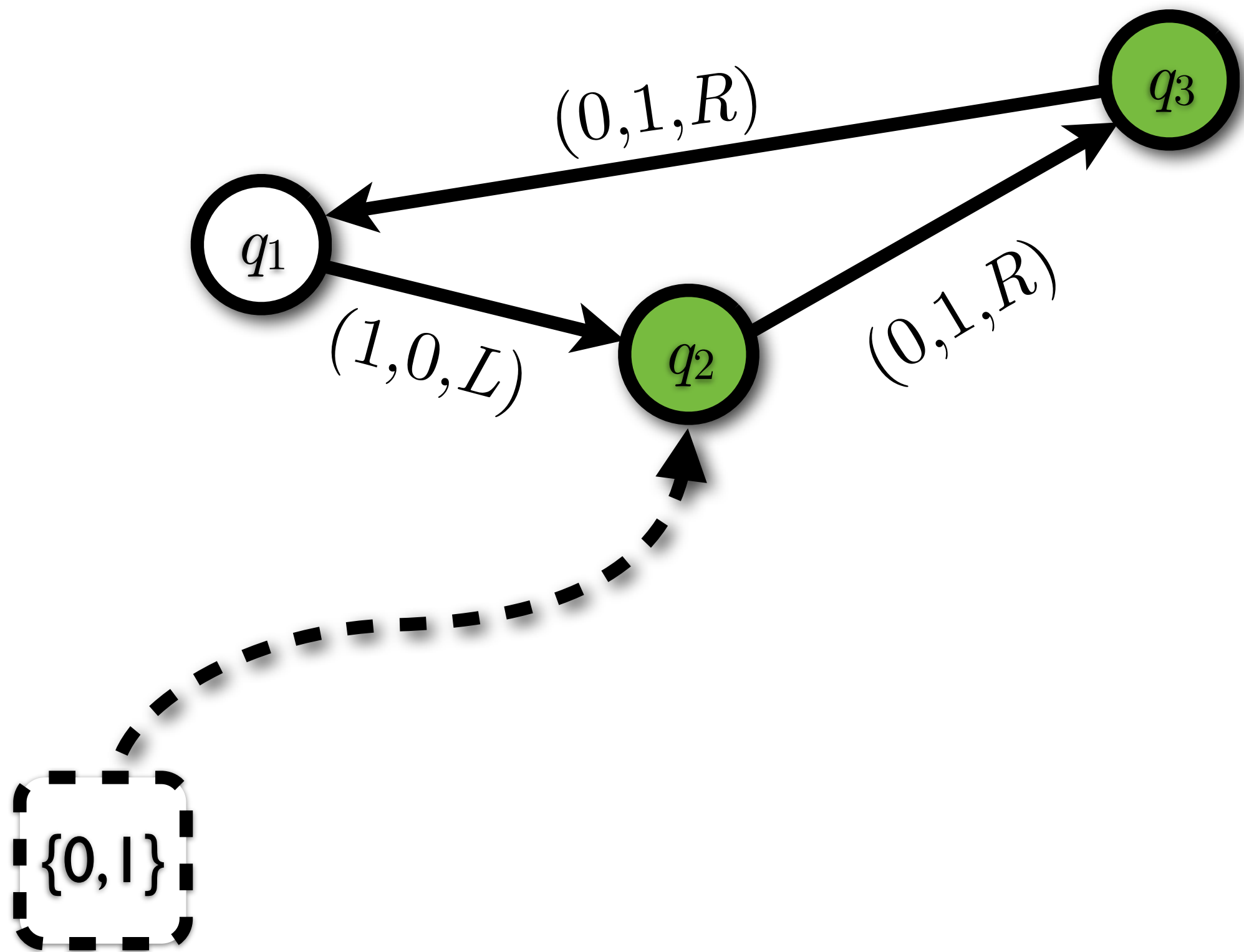
How to approximate?

Make it finite!









Let's do it for RTL.

What is static program analysis?

[\[article index\]](#) [\[email me\]](#) [\[@mattmight\]](#) [\[+mattmight\]](#) [\[rss\]](#)

The halting problem asks whether the execution of a specific program for a given input will terminate.

The halting problem is famous for being undecidable.

That is, no algorithm can solve it for all programs and all inputs.

This complicates any attempt to predict program behavior: we can make predicting almost any program behavior equivalent to predicting the termination of a nearly identical program.

Static analyses are algorithms that do their best to defy the undecidability of the halting problem: they attempt to predict program behavior.

Predicting program behavior enables program optimization, security audits, automatic parallelization and, if accurate enough, correctness verification.

$\langle \text{prog} \rangle ::= \langle \text{stmt} \rangle \dots$

$\langle \text{stmt} \rangle ::=$

- $\langle \text{label} \rangle :$
- $\quad \textbf{goto } \langle \text{label} \rangle ;$
- $\quad \langle \text{var} \rangle := \langle \text{exp} \rangle ;$
- $\quad \textbf{if } \langle \text{exp} \rangle \textbf{ goto } \langle \text{label} \rangle ;$

$\langle \text{exp} \rangle ::=$

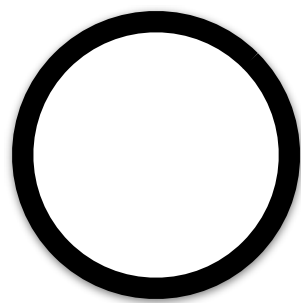
- $\langle \text{exp} \rangle + \langle \text{exp} \rangle$
- $\langle \text{exp} \rangle * \langle \text{exp} \rangle$
- $\langle \text{exp} \rangle = \langle \text{exp} \rangle$
- $\langle \text{int} \rangle$
- $\langle \text{var} \rangle$

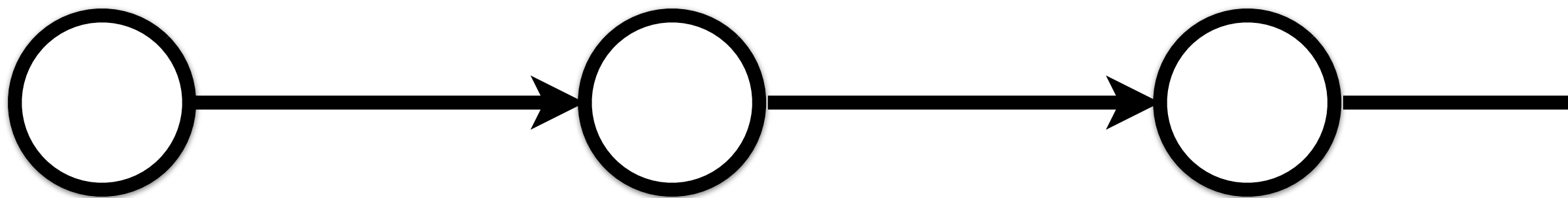
$\langle \text{prog} \rangle ::= \langle \text{stmt} \rangle \dots$

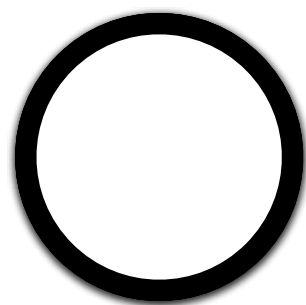
$\langle \text{stmt} \rangle ::=$ $(\mathbf{label} \ \langle \text{label} \rangle)$
 $|$ $(\mathbf{goto} \ \langle \text{label} \rangle)$
 $|$ $(\mathbf{:=} \ \langle \text{var} \rangle \ \langle \text{exp} \rangle)$
 $|$ $(\mathbf{if} \ \langle \text{exp} \rangle \ \mathbf{goto} \ \langle \text{label} \rangle)$

$\langle \text{exp} \rangle ::=$ $(+ \ \langle \text{exp} \rangle \ \langle \text{exp} \rangle)$
 $|$ $(* \ \langle \text{exp} \rangle \ \langle \text{exp} \rangle)$
 $|$ $(= \ \langle \text{exp} \rangle \ \langle \text{exp} \rangle)$
 $|$ $\langle \text{int} \rangle$
 $|$ $\langle \text{var} \rangle$









```
(struct state {stmts env})
```

stmts = stmt*

env = var #=> integer

`inject : prog -> state`

`step : state -> state`

```
; stmt-map : label => stmt*  
(define stmt-map (make-hasheq))
```

preprocess : prog -> void

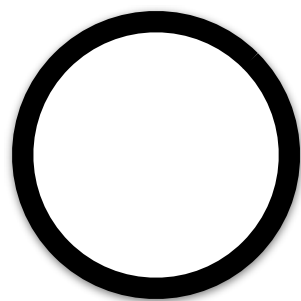
```
(define (preprocess stmts)
  (match stmts
    [(cons `(label ,label) rest)
     (hash-set! stmt-map label stmts)
     (preprocess rest)]

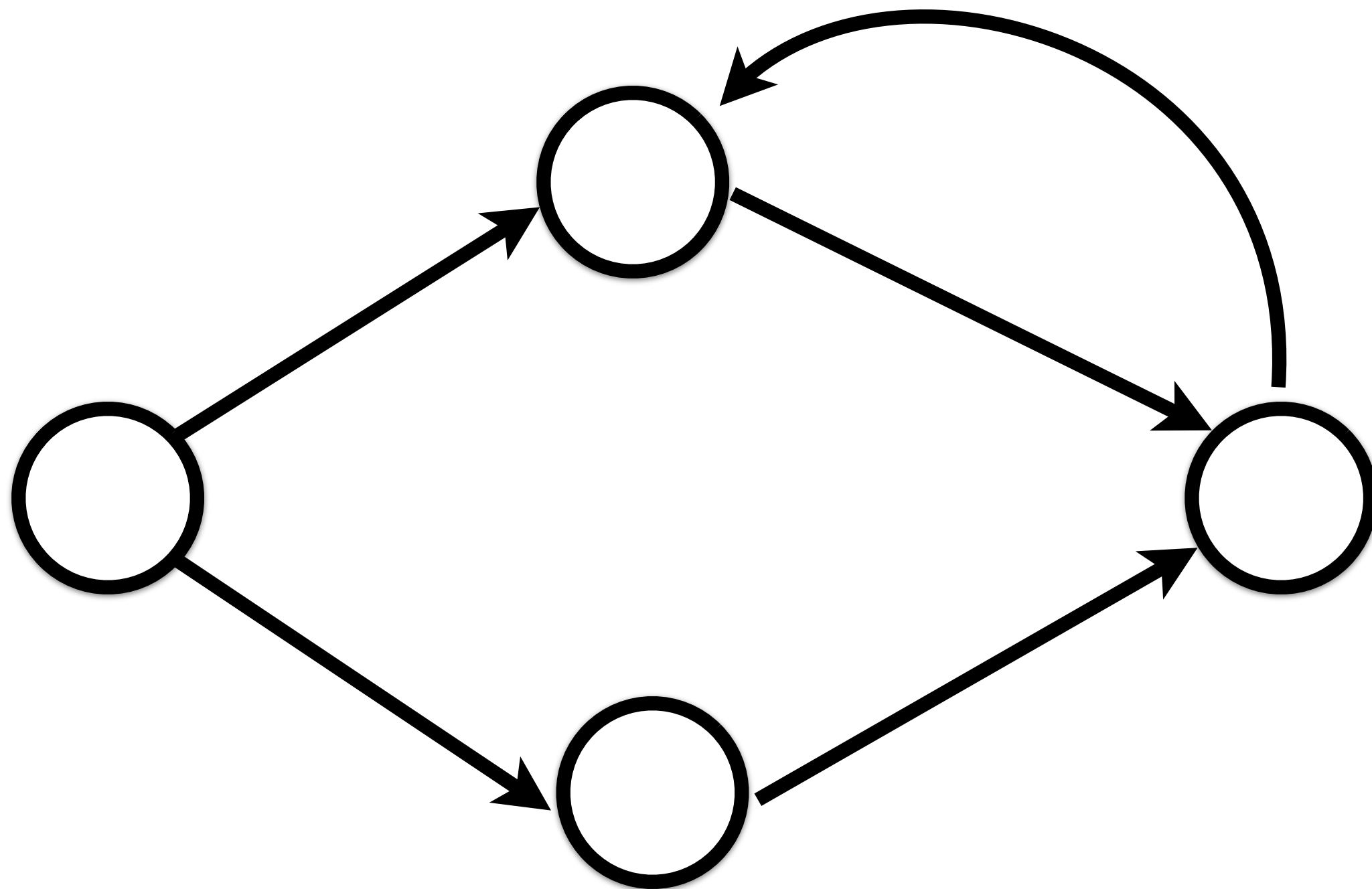
    [(cons _ rest)
     (preprocess rest)]

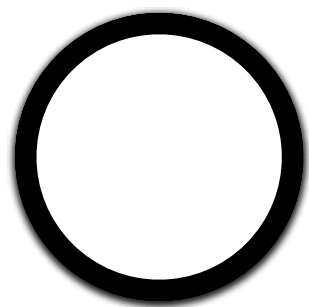
    ['()
     (void)]))
```

Make it finite!









```
(struct state^ { stmts^ env^})
```

$\text{stmts}^{\wedge} = \text{stmt}^*$

$\text{env}^{\wedge} = \text{var} \Rightarrow \text{abstract-integer}$

$\text{inject}^{\wedge} : \text{prog} \rightarrow \text{state}^{\wedge}$

$\text{step}^{\wedge} : \text{state}^{\wedge} \rightarrow \text{state}^{\wedge}$

Questions?

matt.might.net

[@mattmight](https://twitter.com/mattmight)