

COGS300

The Turing Machine

Instructor: Márton Sóskuthy

marton.soskuthy@ubc.ca

TAs: Daichi Furukawa · Victoria Lim · Amy
Wang

cogs.300@ubc.ca

Cognition as computation?

The omelette machine

```
// find the ingredients

number_of_eggs = 3;
eggs = find("eggs", number_of_eggs);
milk = find("milk", a_little);
bowl = find("bowl", 1);
salt = find("salt", a_pinch);
pepper = find("pepper", a_pinch);
butter = find("butter", a_knob);
parsley = find("parsley", a_bunch);
pan = find("pan", 1);

// chop the parsley

parsley_size = estimateSize(parsley);
while (parsley_size > 2) {
    parsley = chop(parsley);
    parsley_size = estimateSize(parsley);
}

// crack the eggs

for (int e = 1; e <= number_of_eggs; e++) {
    egg = eggs[e];
    cracked_egg = crack(egg);
    bowl = bowl + cracked_egg;
}
```

```
// add milk, beat the eggs

bowl = bowl + milk;

for (int i = 1; i <= 12; i++) {
    beat(bowl);
}

// melt some butter in a pan

pan = pan + butter;
cook(pan, 30);

// add the eggs, cook for 2 minutes,
// then fold and cook for 1 minute

pan = pan + bowl;
cook(pan, 120);
fold(pan);
cook(pan, 60);

// add salt, pepper, parsley

pan = pan + salt + pepper + parsley;
```

Heavily computational leanings of early cognitive science inspired by:

- **systems of formal logic**
- **Turing machine**
- physical computers

Formal logic

- p : My dog is happy.
- q : It's raining.

$p \rightarrow \neg q$

q

$\therefore \neg p$



Formal logic

- p : My dog is happy.
- q : It's raining.

$p \rightarrow \neg q$

q

$\therefore \neg p$

p	q	$p \rightarrow \neg q$
F	F	T
F	T	T
T	F	T
T	T	F



Formal logic

- p : My dog is happy.
- q : It's raining.

$p \rightarrow \neg q$

q

$\therefore \neg p$

p	q	$p \rightarrow \neg q$
F	F	T
F	T	T
T	F	T
T	T	F



Formal logic

- p : My dog is happy.
- q : It's raining.

$p \rightarrow \neg q$

q

$\therefore \neg p$

p	q	$p \rightarrow \neg q$
F	F	T
F	T	T
T	F	T
T	T	F



Formal logic

syntax: *primitive symbols*, *operator symbols*, *rules of inference*

$r \vdash \neg\neg r$

$(r \rightarrow s) \vdash (\neg s \rightarrow \neg r)$

$\{(r \rightarrow s); r\} \vdash s$

- p : My dog is happy.
- q : It's raining.

$p \rightarrow \neg q$

q

$\therefore \neg p$



Formal logic

syntax: *primitive symbols, operator symbols, rules of inference*

$r \vdash \neg\neg r$ $(r \rightarrow s) \vdash (\neg s \rightarrow \neg r)$ $\{(r \rightarrow s); r\} \vdash s$

- p: My dog is happy.
- q: It's raining.

$p \rightarrow \neg q$

q

$\neg\neg q$

$\neg\neg q \rightarrow \neg p$

$\therefore \neg p$



Formal logic

propositional logic works the same regardless of the **semantic** content of the propositions!

- p : I'm happy.
- q : Someone's playing a bagpipe.

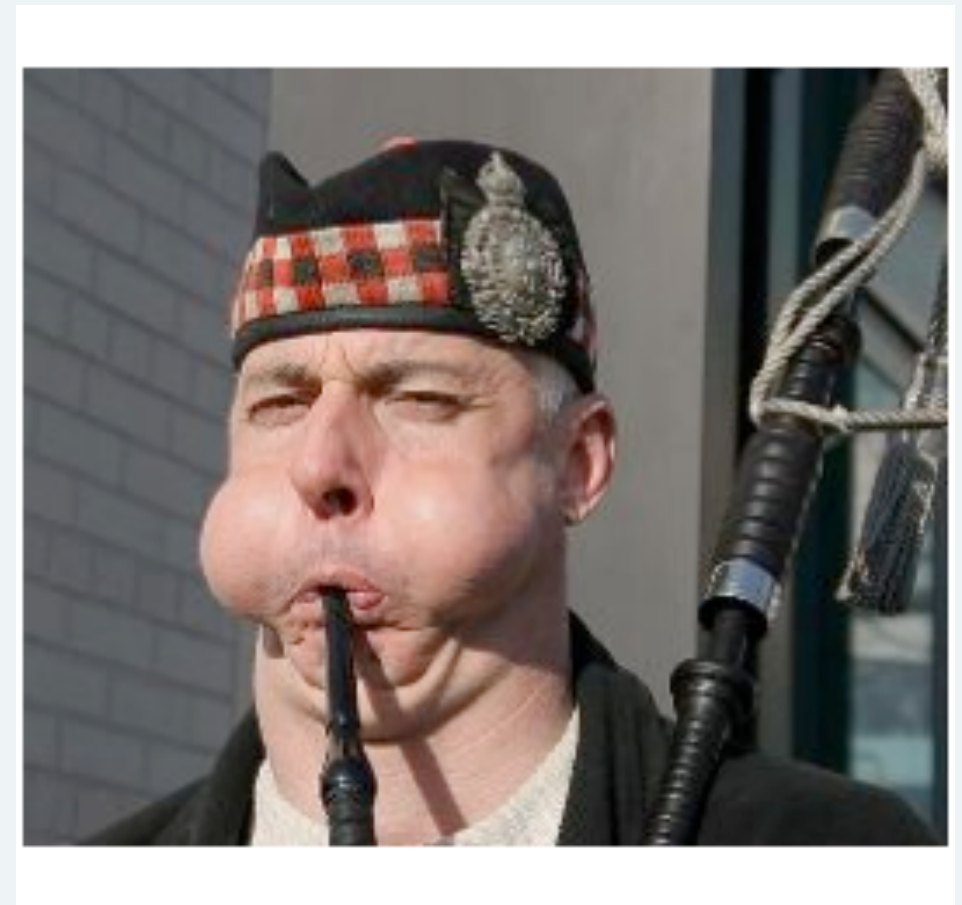
$p \rightarrow \neg q$ (sorry, bagpipe players!)

q

$\neg \neg q$

$\neg \neg q \rightarrow \neg p$

$\therefore \neg p$



Formal logic

- purely syntactic operations \rightarrow preservation of truth (i.e. semantics?)
- in some sense, we could view this system as a model of “reasoning”
- though logical fallacies are a staple of human reasoning...

The long-awaited Turing Machine

- simple problem: counting in (reverse) binary

$$1000 = 1$$

$$0100 = 2$$

$$1100 = 3$$

$$0010 = 4$$

$$1010 = 5$$

$$0110 = 6$$

$$1110 = 7$$

$$0001 = 8$$

The long-awaited Turing Machine

- algorithm 1: start at #
→ move right
0 → 1, done!
1 → 0, move right

#0000... = 0

#1000... = 1

#0100... = 2

#1100... = 3

#0010... = 4

#1010... = 5

#0110... = 6

#1110... = 7

#0001... = 8

The long-awaited Turing Machine

- algorithm 2: start anywhere after (or on) the first 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**

$$10000\dots = 0$$

$$11000\dots = 1$$

$$10100\dots = 2$$

$$11100\dots = 3$$

$$10010\dots = 4$$

$$11010\dots = 5$$

$$10110\dots = 6$$

$$11110\dots = 7$$

$$10001\dots = 8$$

The long-awaited Turing Machine

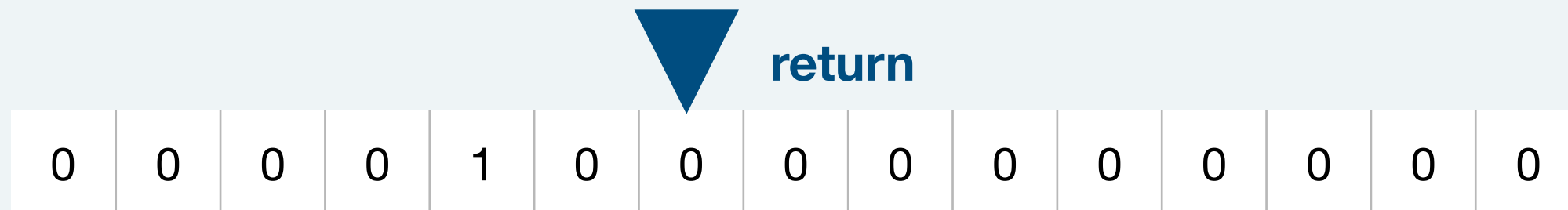
- algorithm 2: start anywhere after the first 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

- algorithm 2: start anywhere after the first 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**


$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**

machine table

return



0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

The long-awaited Turing Machine

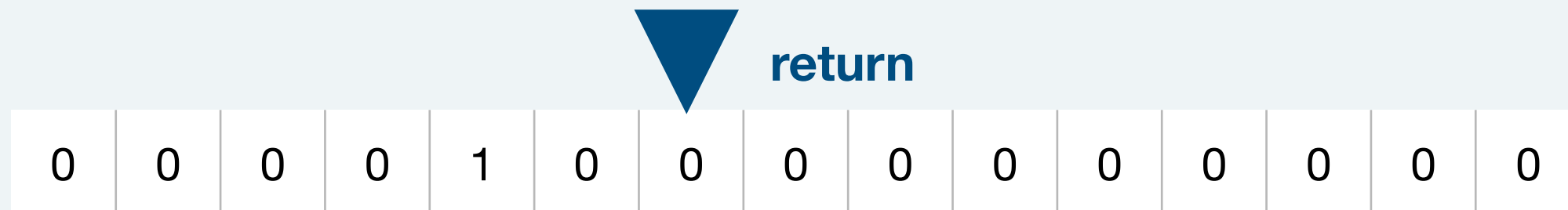
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

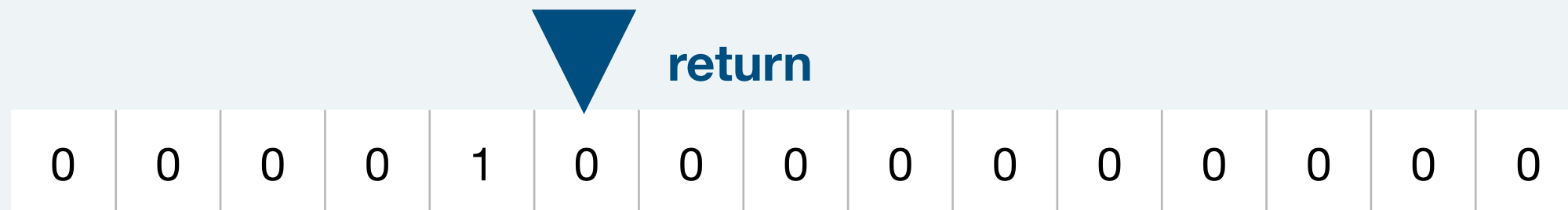
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

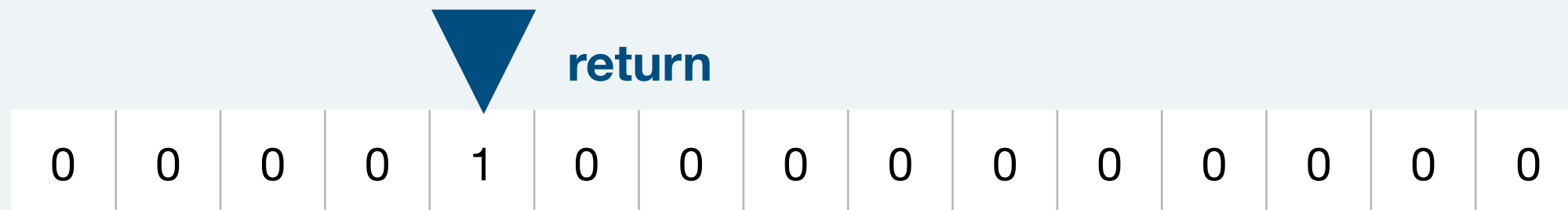
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

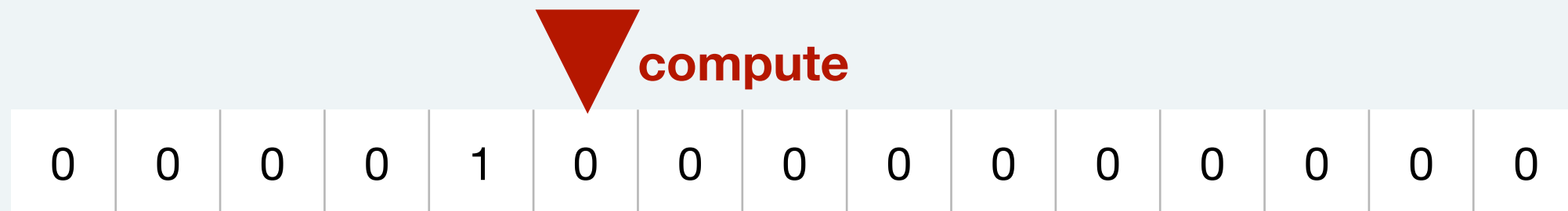
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: 0 → 1, move left, **return**

1 → 0, move right, **compute**

return: 0 → 0, move left, **return**

1 → 1, move right, **compute**



The long-awaited Turing Machine

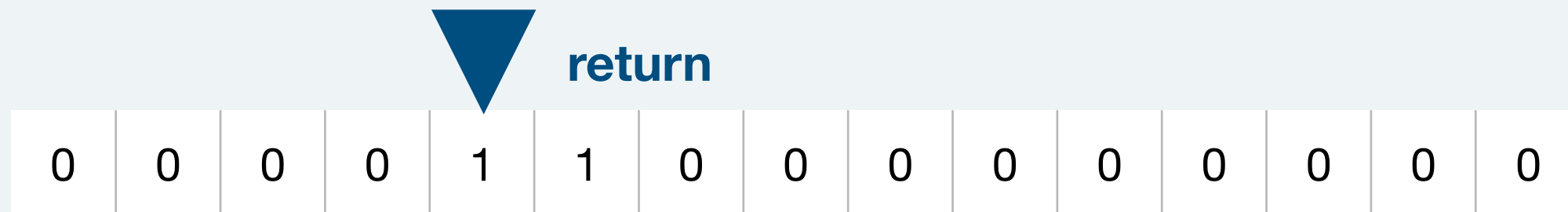
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

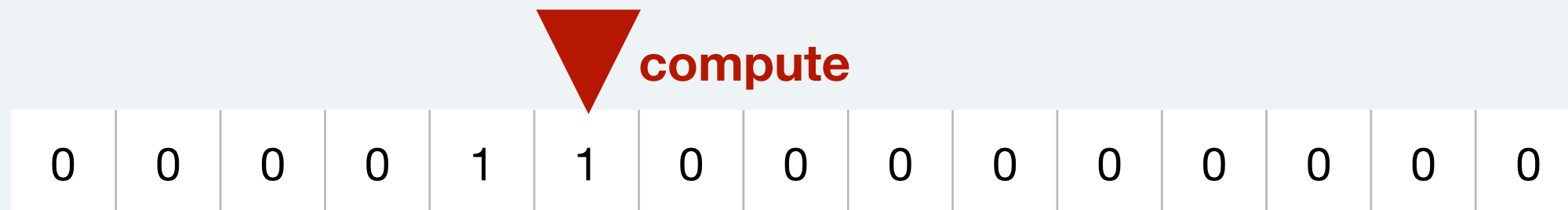
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: 0 → 1, move left, **return**

1 → 0, move right, **compute**

return: 0 → 0, move left, **return**

1 → 1, move right, **compute**



The long-awaited Turing Machine

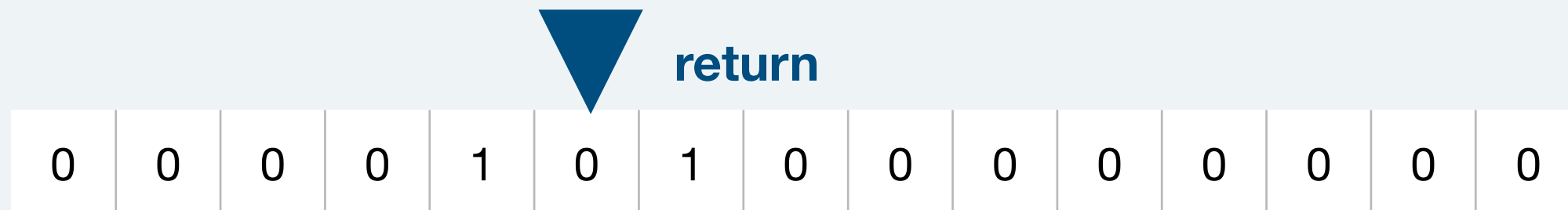
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The long-awaited Turing Machine

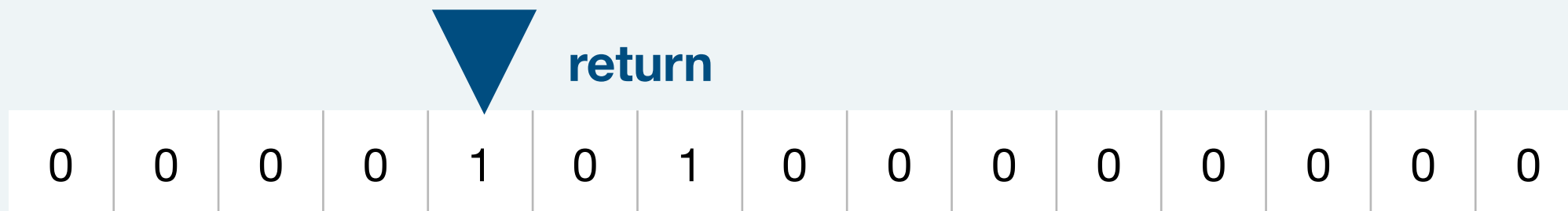
- algorithm 2: if tape is empty, start anywhere after the initial 1 in **return**

compute: $\underline{0} \rightarrow 1$, move left, **return**

$\underline{1} \rightarrow 0$, move right, **compute**

return: $\underline{0} \rightarrow 0$, move left, **return**

$\underline{1} \rightarrow 1$, move right, **compute**



The Universal Turing Machine

- Church-Turing thesis: *~ a function is algorithmically **computable** iff it can be computed by a Turing Machine*
- a (Turing complete) **computer**: a device that can implement algorithms for any computable function

Is the human brain a **computer**?

Applications of TMs?

<http://maximecb.github.io/Turing-Drawings/>

The Turing test

- a proposed threshold for identifying something/someone as an intelligent agent
 - human + hidden conversational partner
 - ongoing open-ended conversation
 - human thinks it's talking to another human: intelligent agent!

The Turing test

- a proposed threshold for identifying something/someone as an intelligent agent
 - human + hidden conversational partner
 - ongoing open-ended conversation
 - human thinks it's talking to another human: intelligent agent!



focus on language / verbal behaviour

The Turing test

- a proposed threshold for identifying something/someone as an intelligent agent
 - human + hidden conversational partner
 - ongoing open-ended conversation
 - human thinks it's talking to another human: intelligent agent!



focus on surface behaviour

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

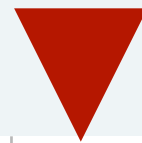
seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move right, s to end

$\underline{1} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 0$, move right, s to end



seek to end

0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end:	<u>0</u> → 1, move left, s to start	<u>1</u> → 1, move right, s to end
seek to start:	<u>0</u> → 0, move right, del 1st	<u>1</u> → 1, move left, s to start
delete 1st 1:	<u>0</u> → 0, move right, del 1st	<u>1</u> → 0, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

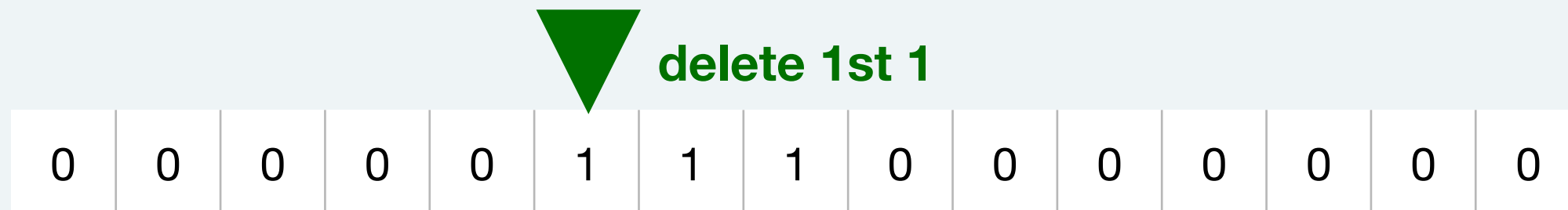
$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

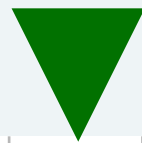
$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



delete 1st 1

0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

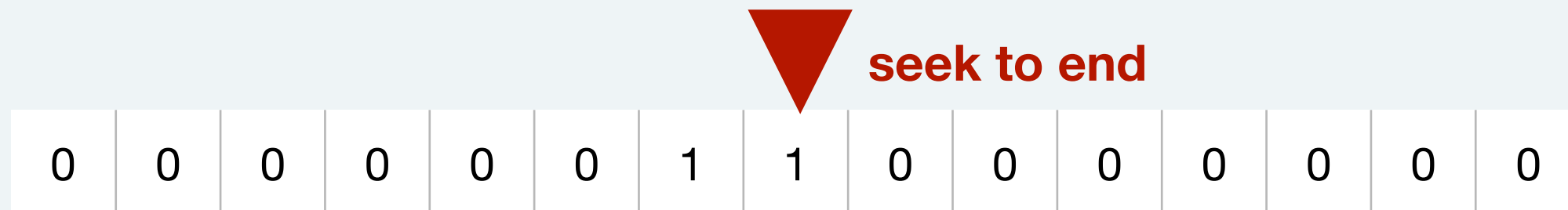
$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

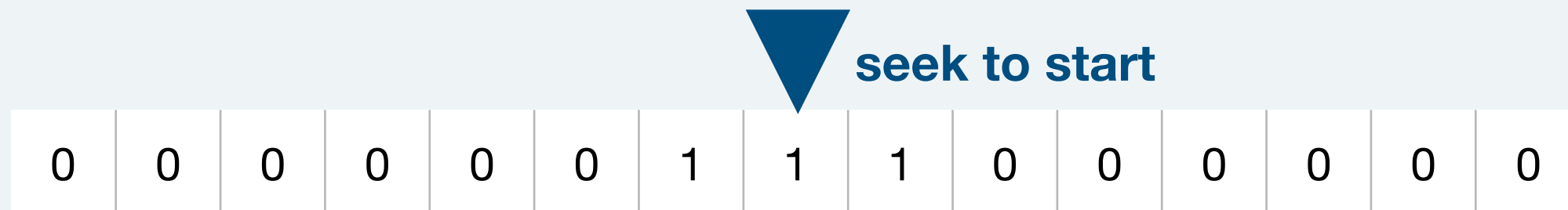
$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



seek to start

0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

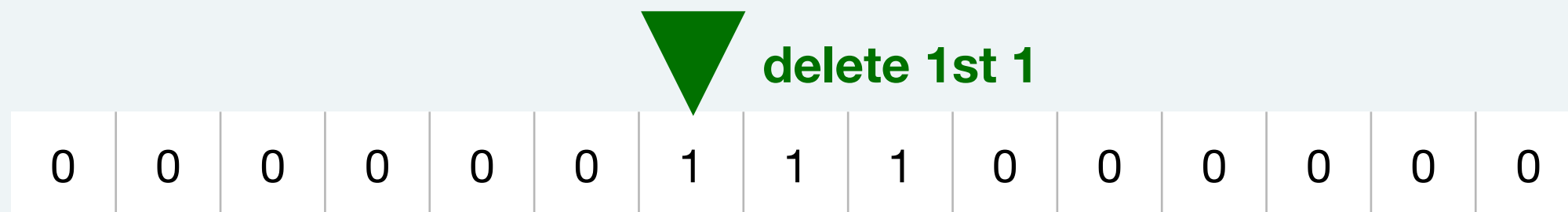
$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end



Turing Machines practice

seek to end: $\underline{0} \rightarrow 1$, move left, s to start

$\underline{1} \rightarrow 1$, move right, s to end

seek to start: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 1$, move left, s to start

delete 1st 1: $\underline{0} \rightarrow 0$, move right, del 1st

$\underline{1} \rightarrow 0$, move right, s to end

