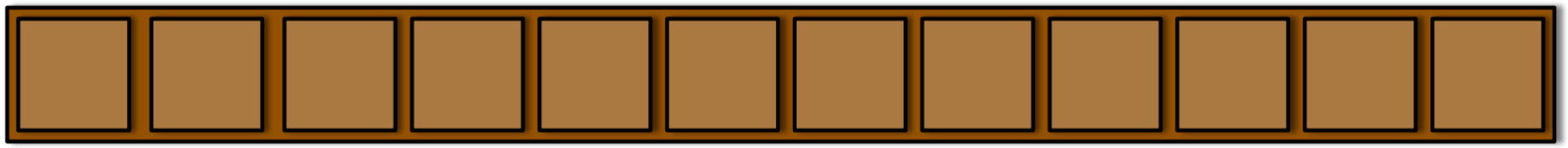
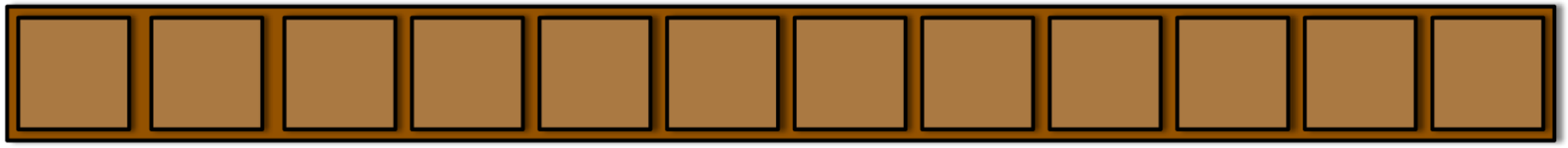
Queen Mary University of London

institute of CODING



The Chocolate Turing Machine



Paul Curzon
Queen Mary University of London

Twitter: @cs4fn

Ask questions via chat

CS4FN

What does a computer look like?

- It could look like anything (including piles of chocolate)

- Its what they do that is more interesting!

Computers can look like anything

- Its what they do and
- how they do it that matters!

Queen Mary University of London

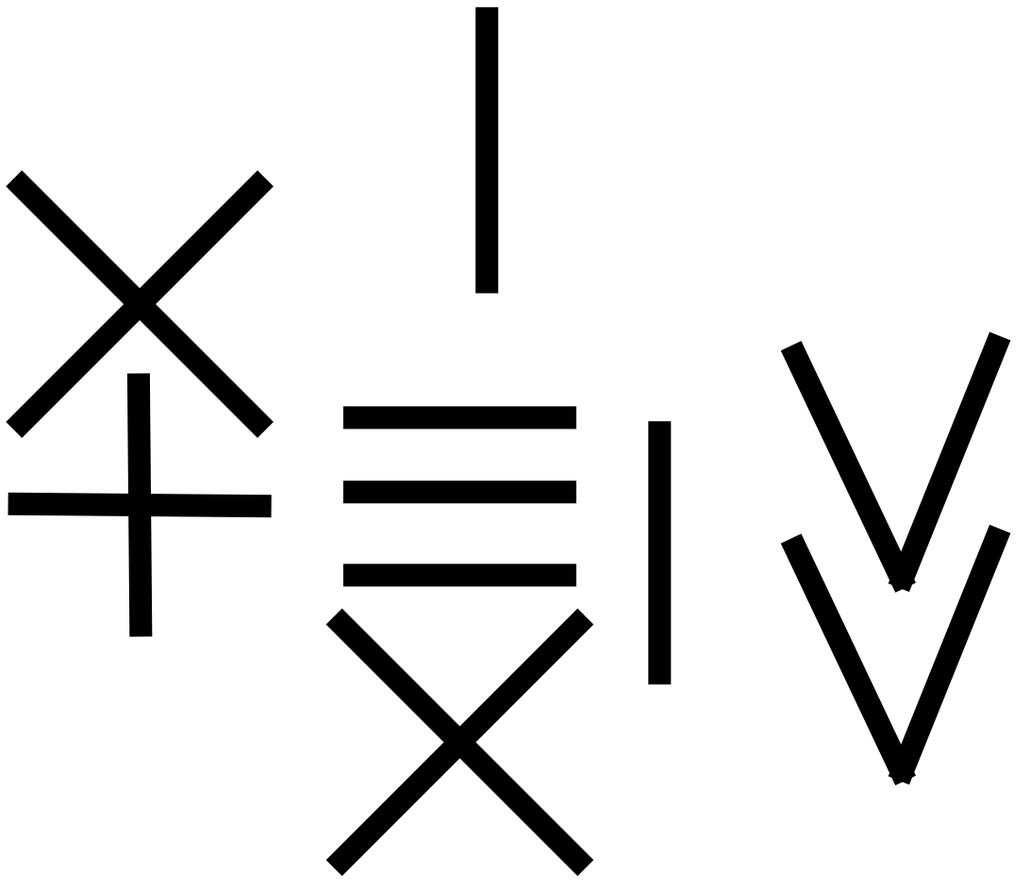
CS4FN

institute of CODING

What does a computer do?

- It does **computation**
- All that really means is it **manipulates symbols**
- A computer takes symbols and replaces them with others
- Over and over and over again
- We (humans) give the symbols meaning

Here are some symbols



- I can invent and follow rules that say how I change patterns of them into new patterns
- that is all computers do!

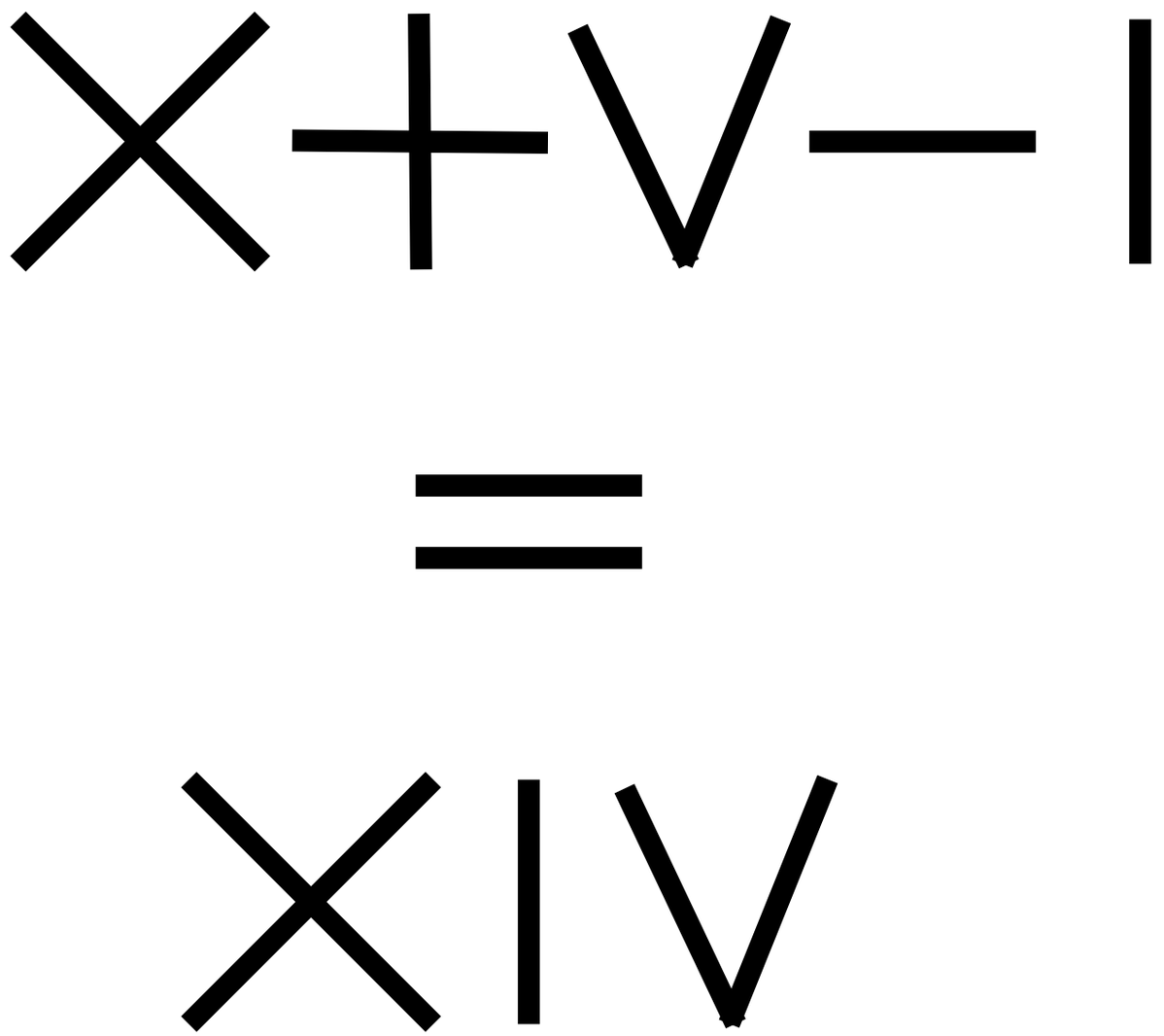
- We give symbols meaning

Queen Mary University of London

CS4FN

institute of CODING

Here are some symbols



- I can invent and follow rules that say how I change patterns of them into new patterns

- that is all computers do!

The same thing using different symbols and different rules

10 plus 5 minus 1

= 14

𓏺	1	𓎆	10
𓏺𓏺	2	𓎆𓎆	20
𓏺𓏺𓏺	3	𓎆𓎆 𓎆	30
𓏺𓏺𓏺𓏺	4	𓎆𓎆 𓎆𓎆	40
𓏺𓏺𓏺 𓏺	5	𓎆𓎆𓎆 𓎆𓎆	50
𓏺𓏺𓏺 𓏺𓏺𓏺	6	𓎆𓎆𓎆 𓎆𓎆𓎆	60
𓏺𓏺𓏺𓏺 𓏺𓏺𓏺	7	𓎆𓎆𓎆𓎆 𓎆𓎆𓎆	70
𓏺𓏺𓏺𓏺 𓏺𓏺𓏺𓏺	8	𓎆𓎆𓎆𓎆 𓎆𓎆𓎆𓎆	80
𓏺𓏺𓏺 𓏺𓏺𓏺 𓏺𓏺𓏺	9	𓎆𓎆𓎆 𓎆𓎆𓎆 𓎆𓎆𓎆	90

The Ancient Egyptians used different symbols again for numbers (with different rules)

…and we still use tally symbols to count

By manipulating symbols we are doing calculations and computation

Queen Mary University of London

CS4FN

institute of CODING

Questions?

We are going to see how to make a working computer out of chocolate …

Queen Mary University of London

CS4FN

institute of CODING

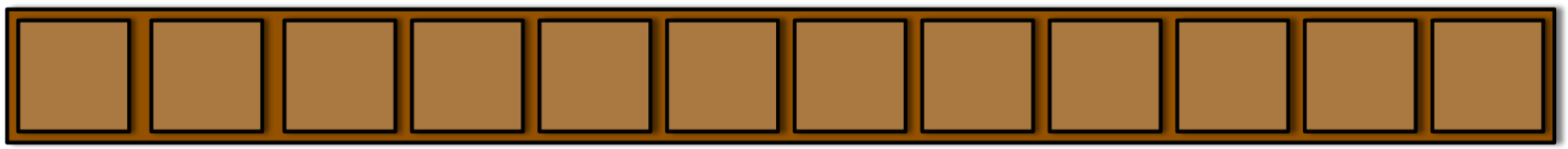
We are going to see how to make a working computer out of chocolate …

…to learn more about “computation” and so explore the limits of what computers can do

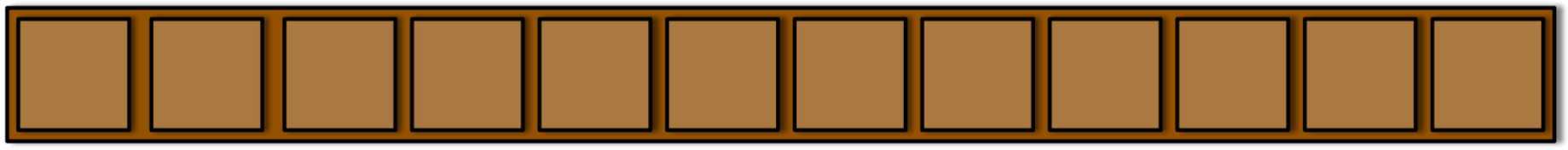
Queen Mary University of London

CS4FN

institute of CODING



WHAT IS A TURING MACHINE?



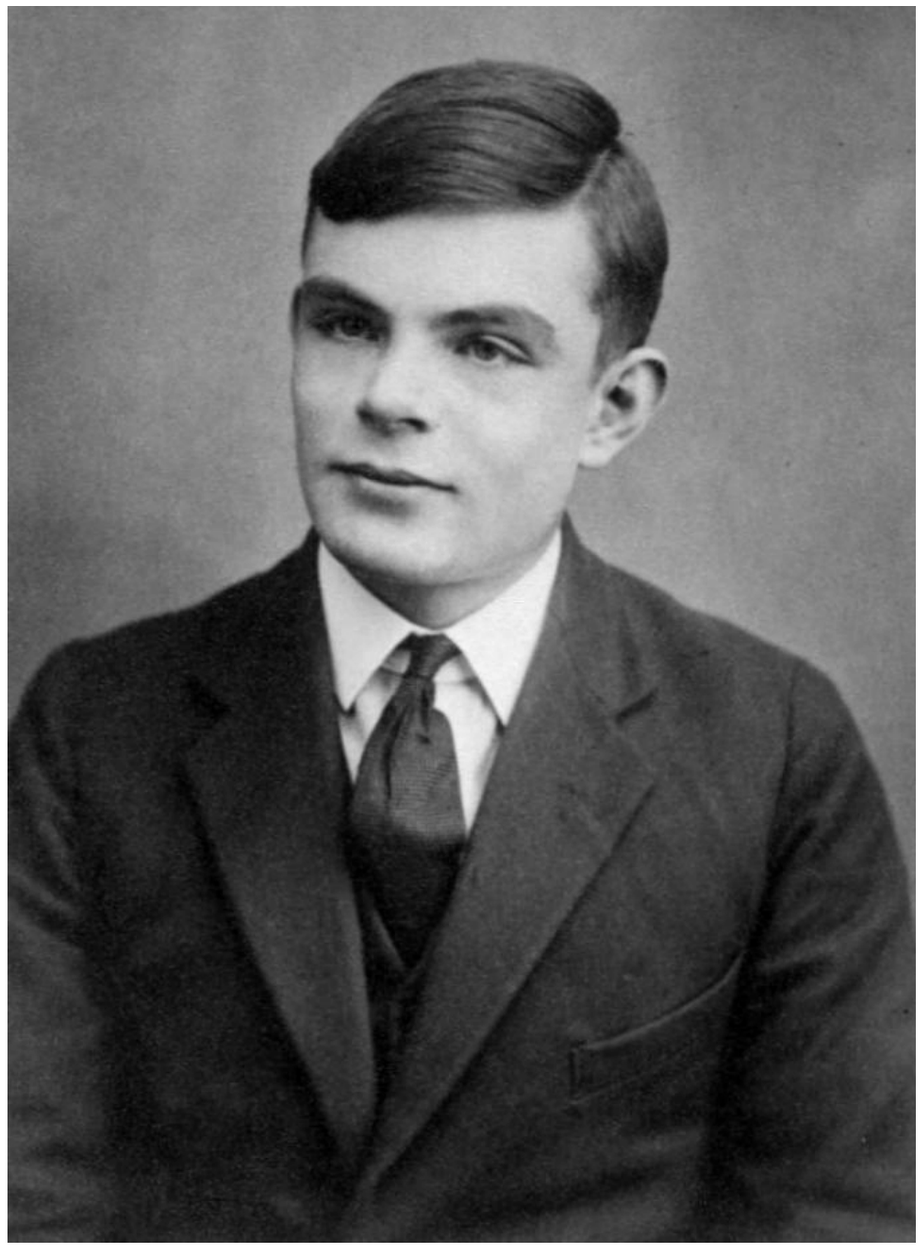
A Turing Machine is a simple "model of computation"

- It **captures the essence of computation**, so what computers can do.

- A Turing machine can do any computation that any computer can do, now or in the future (just not as fast!)

A Turing Machine is a simple “model of computation”

- Invented by Alan Turing in 1936 before any working computer existed
- Each Turing Machine does one fixed job like a gadget with a built in, fixed program
- Turing imagined a human ‘computer’ slavishly working it



Made of chocolate?

- Computation is about manipulating symbols
- Symbols can be anything
- My symbols are chocolate and sweets
- Yours could be made out of Lego stormtroopers…or coins… or bits of paper …

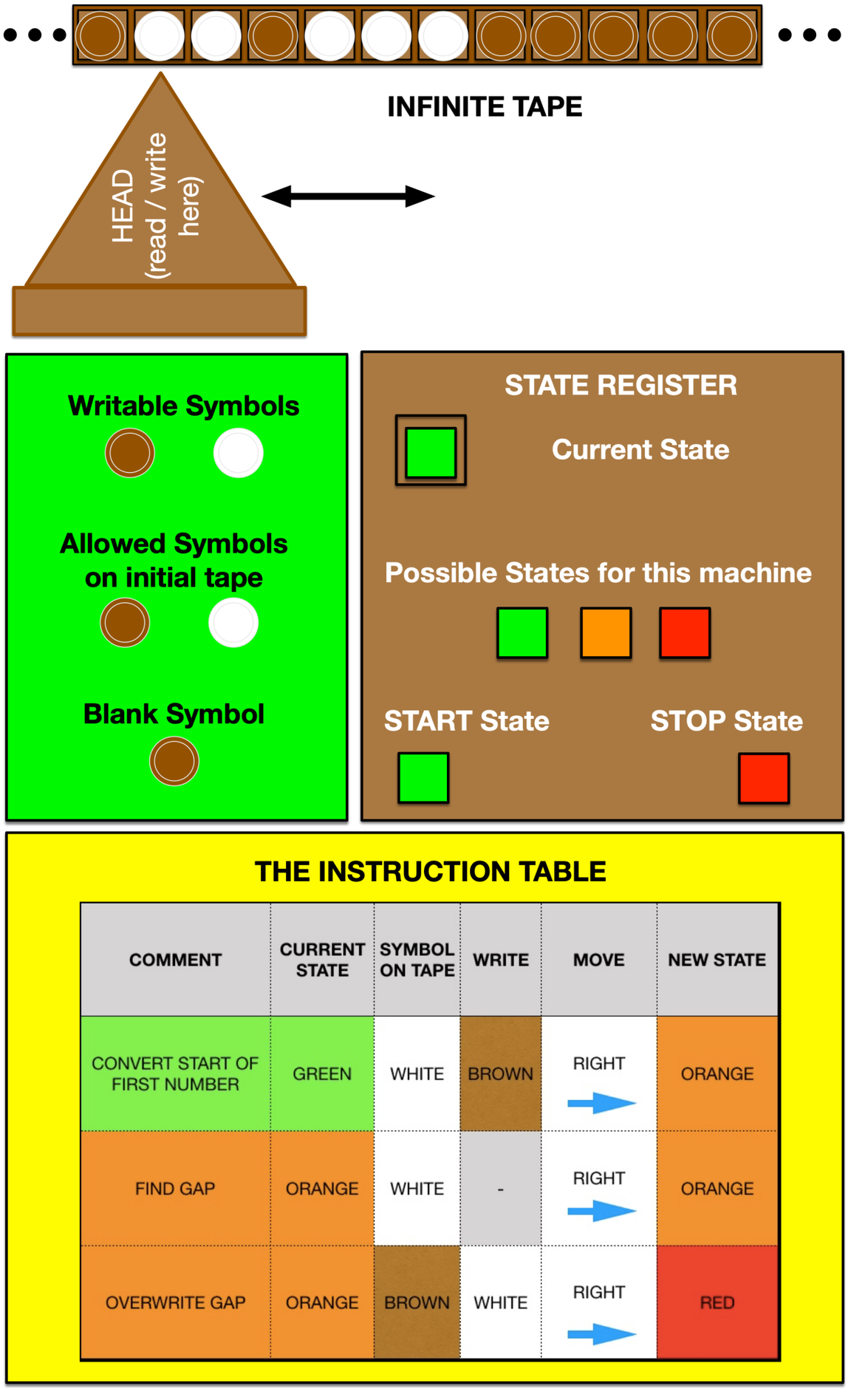
institute of CODING

CS4FN

What does a Turing Machine consist of?

Mine is made of chocolate…

Queen Mary University of London



Activity 1:
Let’s build one…
follow along creating your own

With chocolate and sweets like me if you have them,
if not you can just use paper

Queen Mary University of London

CS4FN

institute of CODING

Questions?

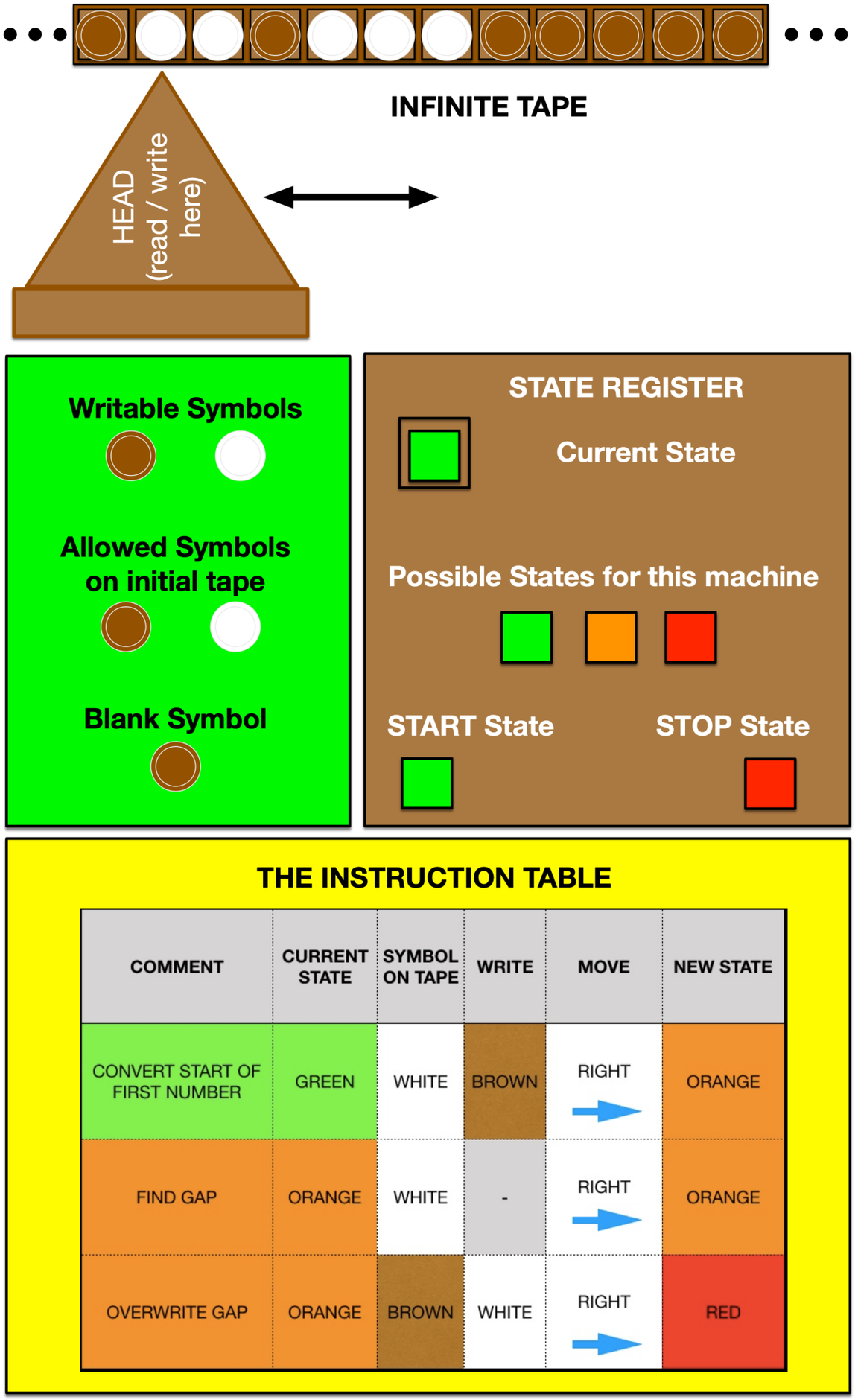
institute of CODING

CS4FN

What does a Turing Machine consist of?

Made of chocolate...

Queen Mary University of London



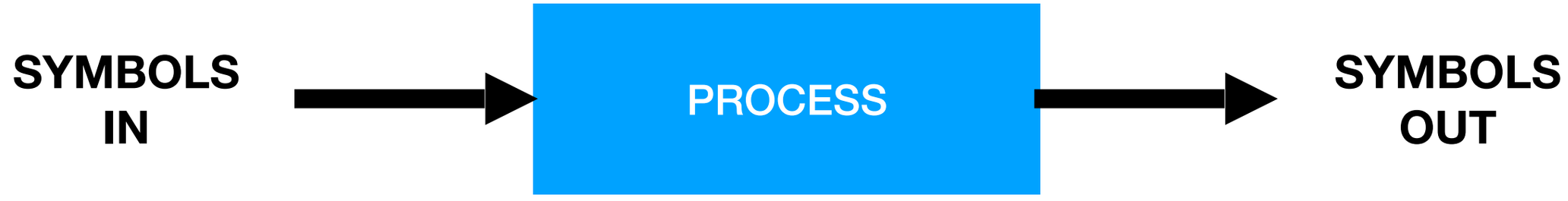
Overview: It consists of…

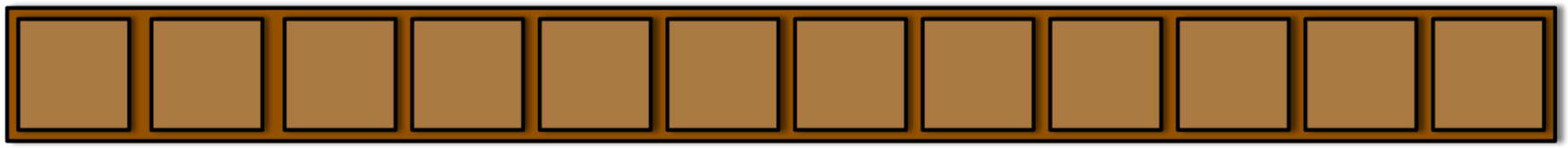
- An infinite tape (its **memory**) containing symbols
- Symbols that can be written on to the tape (eg 0, 1)
- Symbols that can appear on the original tape, including a BLANK symbol
- A read/write head (the one point it can change values)
- A finite set of possible states.
- A state register (to keep track of the state)
  - The current state from a fixed set (its “state of mind”)
  - A START state and a STOP state (if the machine enters this state it stops)
- A finite table of instructions (the **program**)

Input-Output

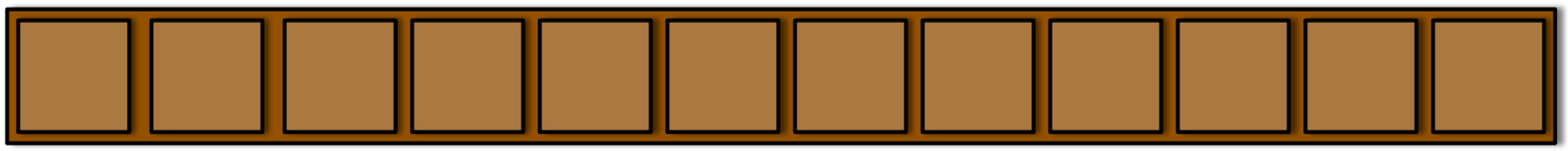
A Turing Machine follows an input-process-output pattern

- INPUT: the initial tape
- PROCESS: the operation of the machine
- OUTPUT: the final tape once it STOPs





IN PRACTICE



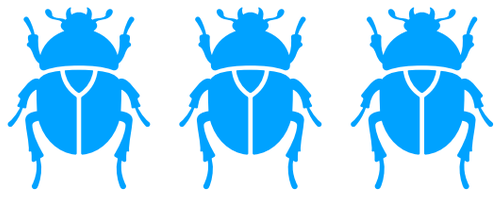
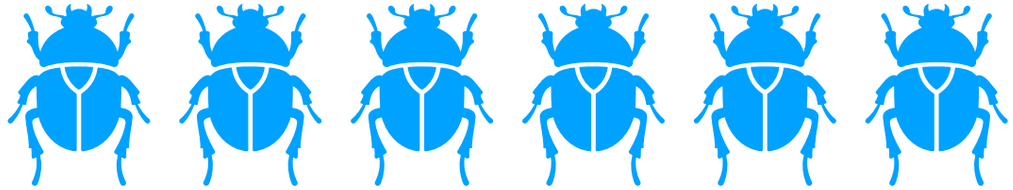
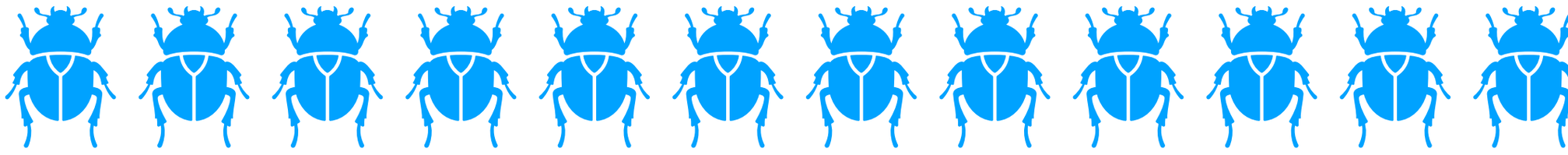
Number Representation

- We can use ANY number representation we like
  - particular groups of symbols to mean each number (and rules for counting, adding etc)
- To keep our first program simple we will use the earliest number representation used by ancient shepherds to count goats! … UNARY
  - eg 5 is represented as ooooo (surrounded by blanks)
- A more sophisticated program could use binary, base 10 or roman numerals - just different tape symbols.

Activity 2: Write as Unary (in a symbol of your choice)

- 3
- 6
- 11
- 99

Solutions

- 3 
- 6 
- 11 
- 99 

Oh no I've run out of space...

Questions?

To ADD
two unary numbers…

- eg with a starting tape (and head pointing at first o):

…_oo_ooo_ _ _ …

1. Blank the first symbol (at start of first number)
2. Find the gap between the numbers
3. Write an o symbol in the gap between the numbers

- Assumes the numbers are separated by a single blank

An Instruction Table

	IF STATE NOW THEN		ACTIONS TO TAKE		
COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
BLANK START OF FIRST NUMBER	GREEN	WHITE	BROWN	RIGHT →	ORANGE
FIND GAP	ORANGE	WHITE	-	RIGHT →	ORANGE
OVERWRITE GAP	ORANGE	BROWN	WHITE	RIGHT →	RED

Let’s run it …

(you follow along at home)

First copy this instruction table…

(either by drawing it or taking a screenshot or …)

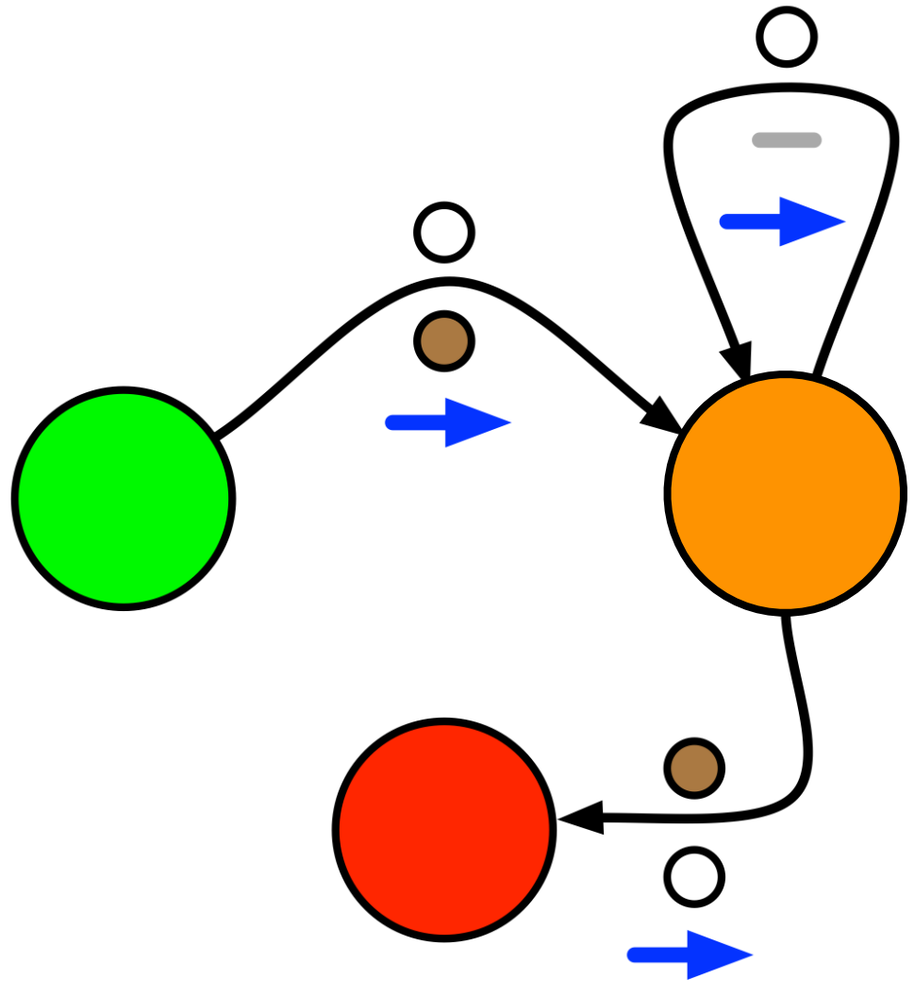
CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
			→	
		–	→	
			→	

Another way to do the same thing: Going to the other end of the number

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
TRAVERSING FIRST NUMBER	GREEN	WHITE	-	RIGHT →	GREEN
	GREEN	BROWN	-	RIGHT →	PURPLE
TRAVERSING SECOND NUMBER	PURPLE	WHITE	-	RIGHT →	PURPLE
	PURPLE	BROWN	-	LEFT ←	BLUE
CONVERTING END OF SECOND NUMBER	BLUE	-	BROWN	LEFT ←	ORANGE
MOVING BACK TO GAP	ORANGE	WHITE	-	LEFT ←	ORANGE
FILL GAP AND STOP	ORANGE	BROWN	WHITE	LEFT ←	RED

State Transition Diagrams

Instruction Tables can be written in pictures as a:
State Transition Diagram



These can be a bit easier to follow…

State transition diagrams (and instruction tables) can be made of sweets too!

Queen Mary University of London

CS4FN

institute of CODING

Questions?

Activity 3

Create a Turing Machine to Add 2 to a unary number

Work out a Turing Machine Instruction Table to add 2 to a unary number. Check it works on your Turing Machine

- It should add it to the right-hand end of the number leaving the number starting in the same place
- Extension …
- The Head should return to the position at the left hand end of the number, where it started.

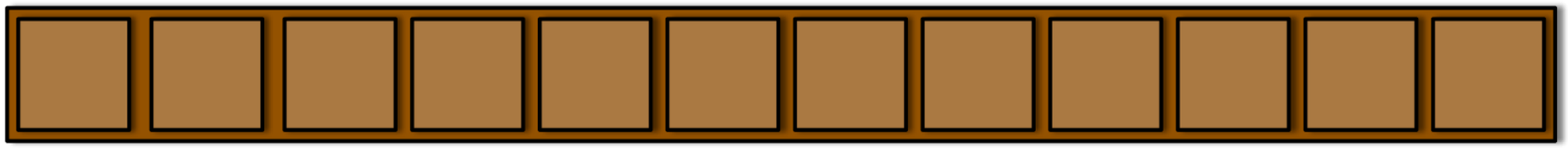
- Here is a reminder of the instruction table for adding two unary numbers to help

CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
			→	
		-	→	
			→	

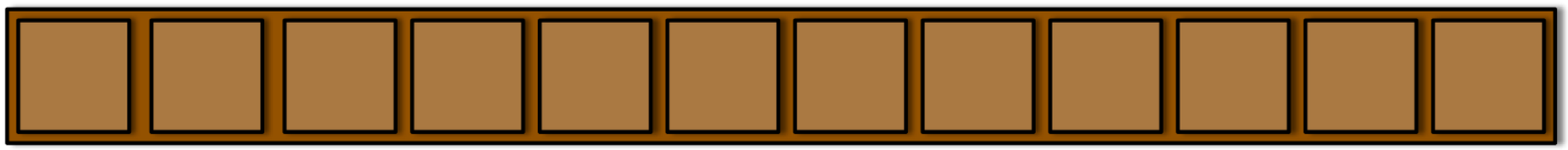
Queen Mary University of London

CS4FN

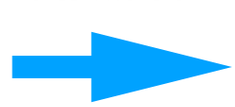
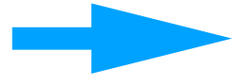
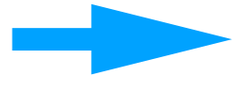
institute of CODING



SOLUTION



An Add 2 Instruction Table

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT 	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT 	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	RIGHT 	RED

An Add 2 Instruction Table

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT →	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT →	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	RIGHT →	RED

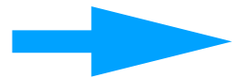
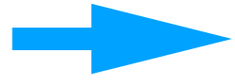
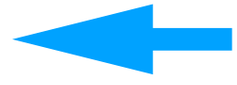
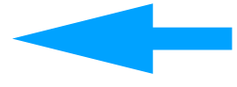
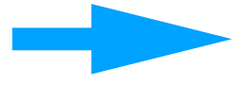
An Add 2 Instruction Table

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT →	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT →	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	RIGHT →	RED

An Add 2 Instruction Table

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT →	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT →	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	RIGHT →	RED

Add 2, returning to the start

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT 	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT 	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	LEFT 	PURPLE
HEAD BACK TO START	PURPLE	WHITE	-	LEFT 	PURPLE
	PURPLE	BROWN	-	RIGHT 	RED

Add 2, returning to the start

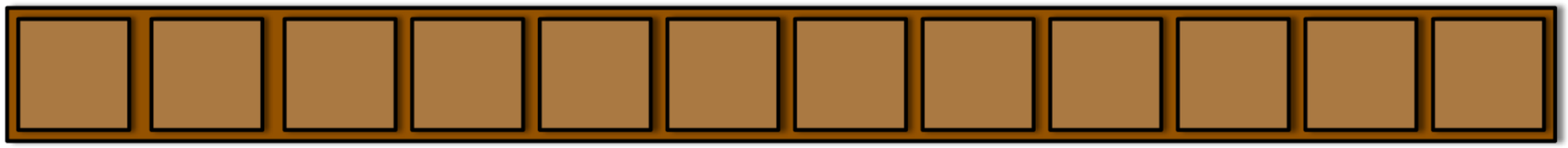
COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	LEFT	PURPLE
HEAD BACK TO START	PURPLE	WHITE	-	LEFT	PURPLE
	PURPLE	BROWN	-	RIGHT	RED

Add 2, returning to the start

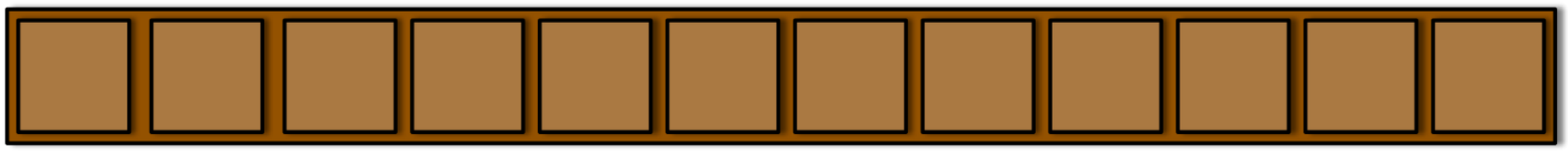
COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	LEFT	PURPLE
HEAD BACK TO START	PURPLE	WHITE	-	LEFT	PURPLE
	PURPLE	BROWN	-	RIGHT	RED

Add 2, returning to the start

COMMENT	CURRENT STATE	SYMBOL ON TAPE	WRITE	MOVE	NEW STATE
MOVE TO END OF NUMBER	GREEN	WHITE	-	RIGHT	GREEN
ADD 1	GREEN	BROWN	WHITE	RIGHT	ORANGE
ADD ANOTHER 1	ORANGE	BROWN	WHITE	LEFT	PURPLE
HEAD BACK TO START	PURPLE	WHITE	-	LEFT	PURPLE
	PURPLE	BROWN	-	RIGHT	RED



THEORY



Queen Mary University of London

CS4FN

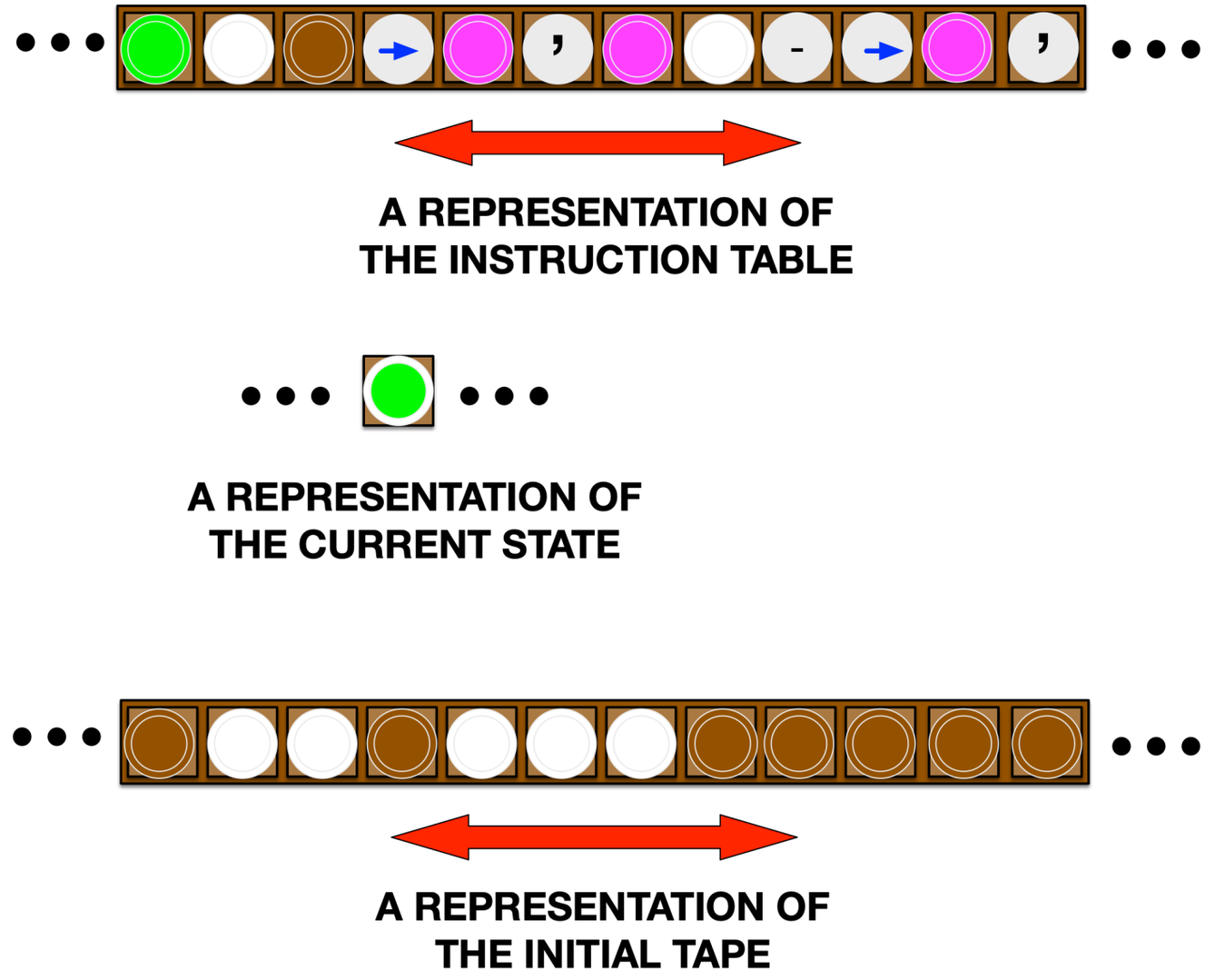
institute of CODING

As powerful as is possible

- Turing Machines can be programmed to do anything that any computer can do
  - They can compute any computation that it is possible for a computer to compute!
- Has **sequence**, **selection** & **repetition**
- You create a bespoke Turing Machine to do the computation you need to be done.

A Universal Turing Machine

- Can create a Turing Machine that can simulate any other Turing Machine
  - put the instruction table on the tape
  - write a simulator instruction table



A fundamental breakthrough

The idea of a Universal Turing Machine is the basis for a

- **general-purpose computer**
  - ie a computer that can do “anything”
- because it is controlled by **a stored program** that tells it what to do.

A fundamental theoretical breakthrough

in computer science

Questions?

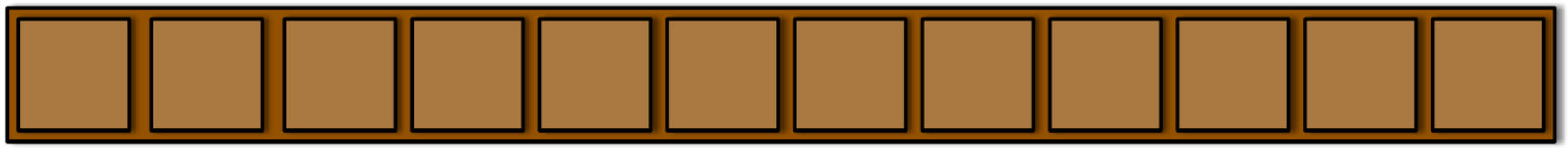
Summary

- A Turing Machine is just a simple model of what it is possible for computation to do.
- Turing Machines can compute anything any real computer can compute.
- You can make a UNIVERSAL Turing Machine that can simulate all other Turing Machines.
    - The description of the Turing Machine is just the data on UNIVERSAL's initial tape.

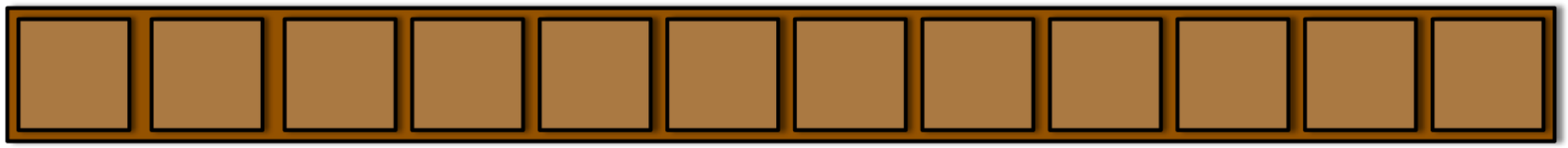
More Activities to try

Create Turing Machines to …

1. Multiply a decimal number by 10
   - Extension: The Head should return to the position at the left hand end of the number, where it started.
2. Go through a traffic light sequence of symbols on the tape
3. Encrypt messages using a substitution cipher
   - Swapping characters in a message on a tape for different ones
4. Decrypt messages using the same substitution cipher as above
5. Count using Egyptian, Roman or other numeral systems



Thank You

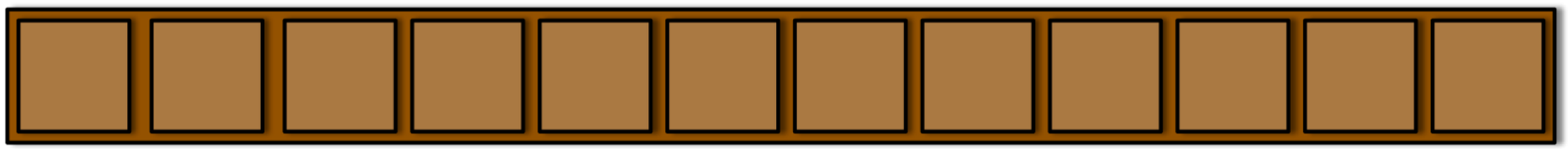


Our free resources at teachinglondoncomputing.org/turingmachine/

Twitter: @cs4fn

Isaac Computer Science at isaaccomputerscience.org/topics/

-> Theory of computation (AQA) -> Turing Machines

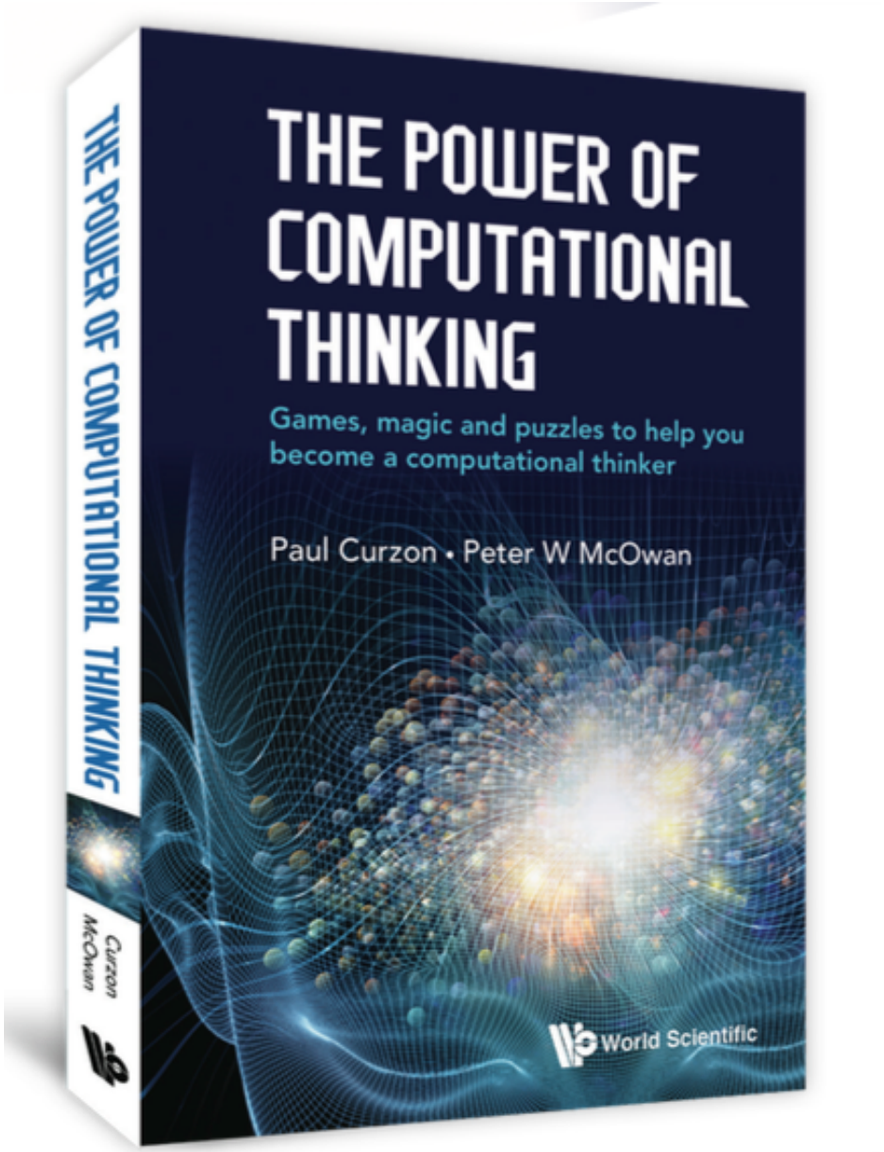


Queen Mary University of London

CS4FN

institute of CODING

If interested in more about computation from a fun point of view…



You might like our book on computational thinking

(the one thing you have to pay for aside from chocolate)

https://www.worldscientific.com/worldscibooks/10.1142/q0054

Order from your local bookshop as they need the help