Micro:bits
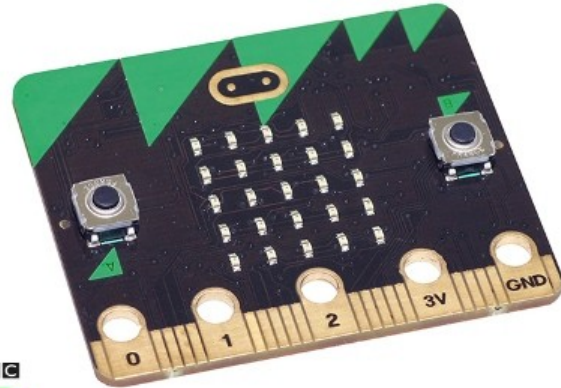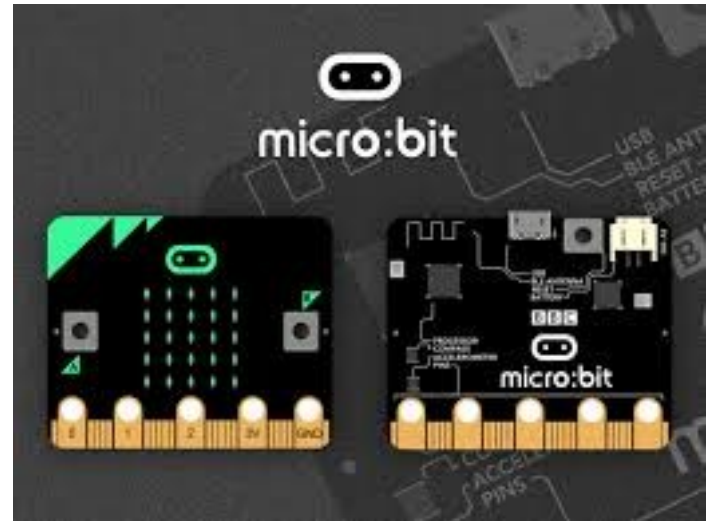
Background info

- Developed in the UK so all students would have access to a tool to learn about computer science
    - Computer science → the study of **how users can actually use a computer**
    - **Clip** to see how they are used

Need for Computer Science

- Computers are important to everyone's lives in the current world
- Computer examples

- Can we use them effectively until we know how they communicate with the user?
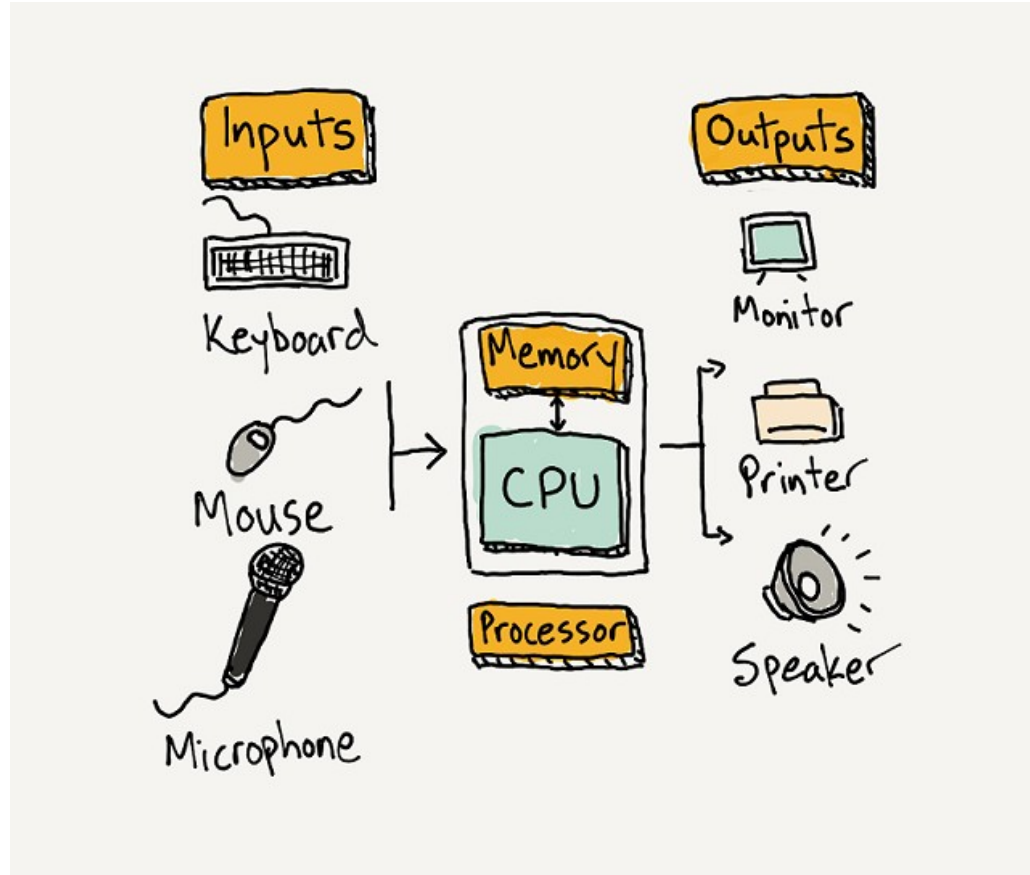
Computers

All computers have four parts:

- **Input**
- **Processor**
- **Output**
- **Memory**

How do these compare to humans?

- Input - Senses
- Output - Speech / Blushing
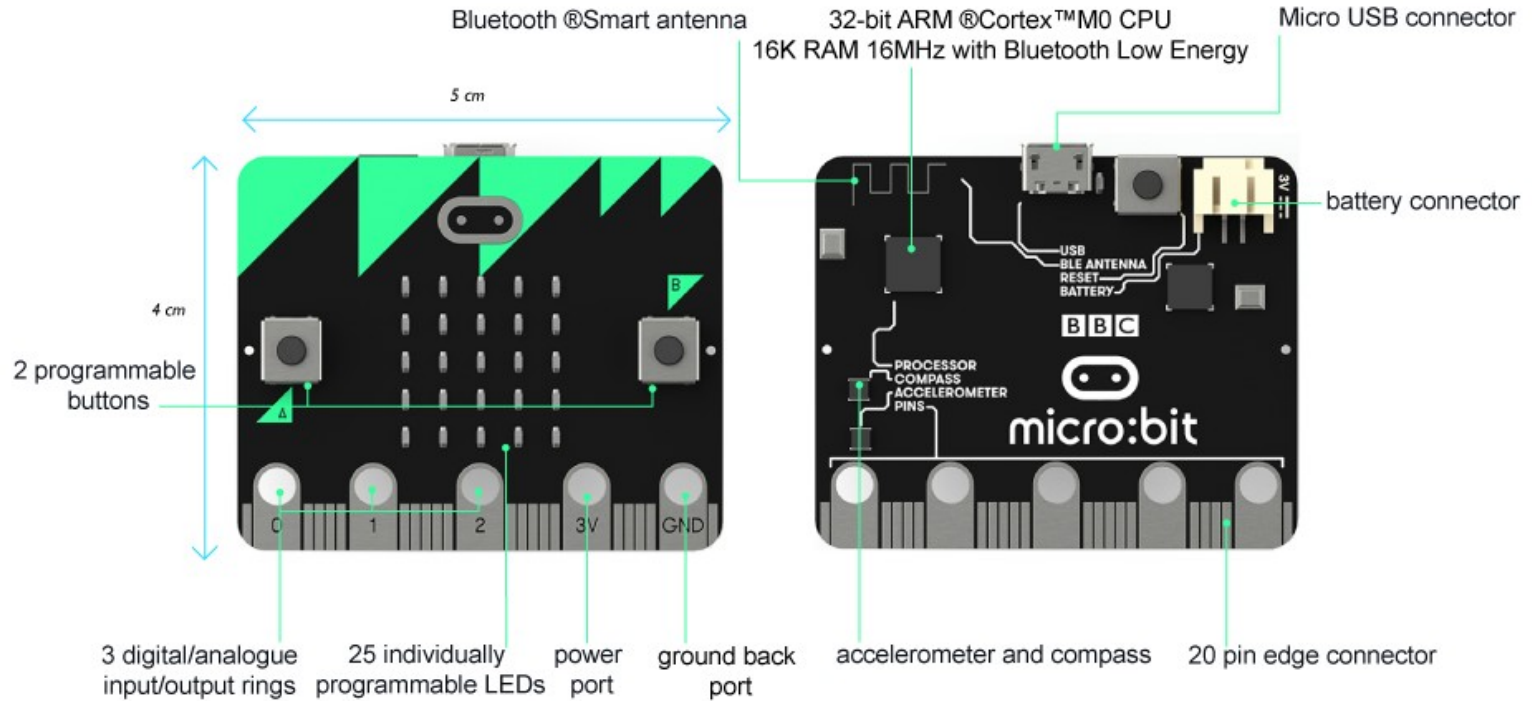- Processor and Memory - Brain

Micro:bit components

Can you locate these parts on the Micro:bit?

- Processor
- Inputs → buttons, pins, accelerometer, compass, light sensor
- Outputs → LED lights, pins, Bluetooth

(Memory is 16K RAM)

Micro:bit components

Bluetooth ®Smart antenna

32-bit ARM ®Cortex™M0 CPU
16K RAM 16MHz with Bluetooth Low Energy

Micro USB connector

battery connector

5 cm

4 cm

B

A

2 programmable
buttons

USB
BLE ANTENNA
RESET
BATTERY

BBC

PROCESSOR
COMPASS
ACCELEROMETER
PINS

micro:bit

0   1   2   3V   GND

3 digital/analogue
input/output rings

25 individually
programmable LEDs

power
port

ground back
port

accelerometer and compass

20 pin edge connector

FRONT

BACK

Computer Science: Algorithms

Computers use **algorithms** to process the information in the form of instructions.

- Precise set of **ordered** instructions.
- Goal is to solve the problem presented.
- Usually accurate (or no point using it).

Example:

Input: Start with 6.

Process (algorithm): Multiply by 5.

Output: 30

Second example:

Input: Spaghetti.

Algorithm:
Open tin.
Put spaghetti into bowl.
Place bowl in microwave.
Close microwave door.
Press 30 seconds.

Output: Warm spaghetti.
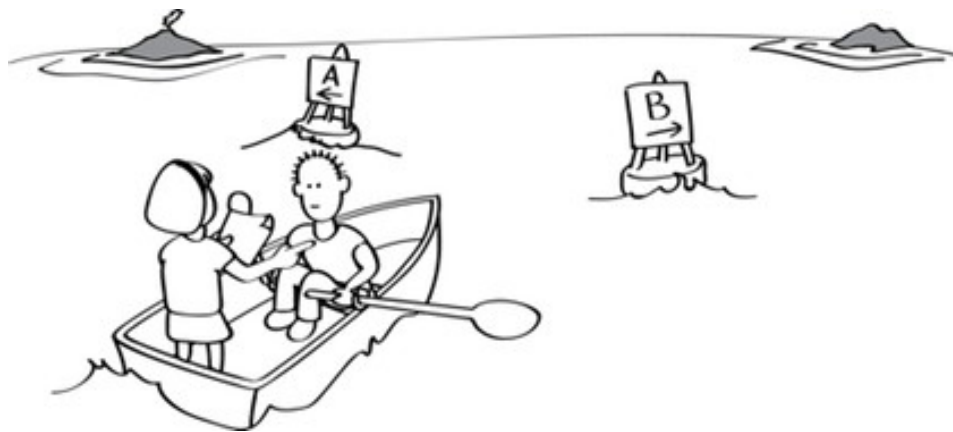
Algorithms

Algorithms can be:

- Finite (yes / no)
- Conditional (and, or, and/or)

Examples:

When the light turns red, you stop.

If a student is well behaved and wearing something purple, they get a cookie.

A doctor prescribes antibiotics if a patient has swollen tonsils but could also prescribe them if the patient had an earache too.

Algorithms activity

Step #1: In groups of 3-4, you'll be given a series of "If" statements.

For example, "If I were hungry…" or "If my cat escaped for the day…"

You need to complete the statement. One answer per blank piece of paper.

Step #2: Shuffle up your answer statements and turn them face down.

Step #3: One "If" statement will be read out. Select the statement at the top of your pile to complete the sentence.
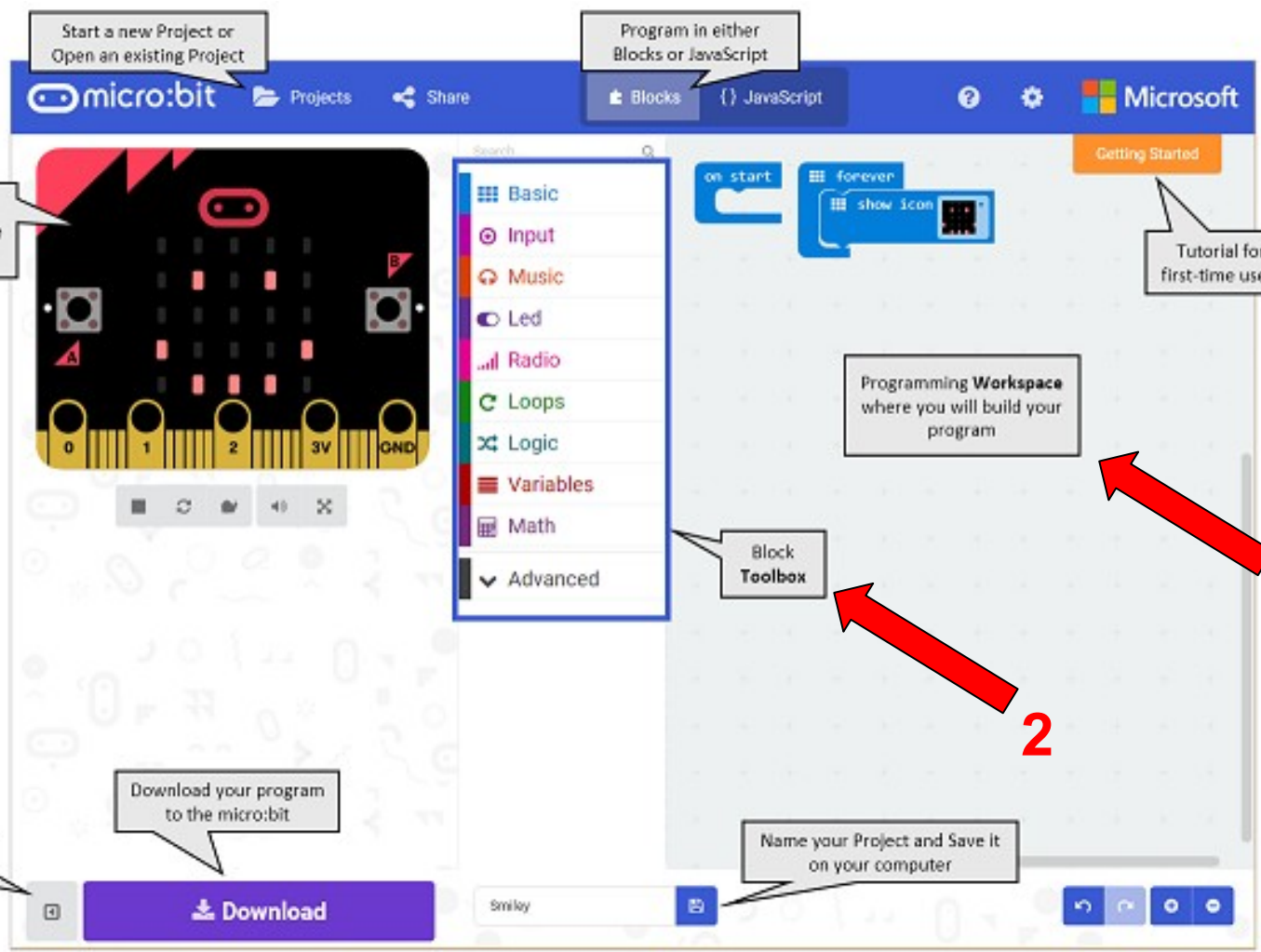
Does this make any sense?? No - which is why algorithms make sure this doesn't happen.

Using the micro:bits coding site

Coding is done online at **https://makecode.microbit.org**

Three main parts:

1. Workspace - where the code is written
2. Toolbox - what you use to write the code
3. Simulator - what the code does to the micro:bit (lets you see what happens before you download the code)

Start a new Project or
Open an existing Project

Program in either
Blocks or JavaScript

Projects   Share   Blocks   {} JavaScript

Microsoft

Getting Started

Simulator shows what
your program will look like
running on a micro:bit

Tutorial for
first-time users

Search

Basic
Input
Music
Led
Radio
Loops
Logic
Variables
Math
Advanced

on start   forever   show icon

Programming **Workspace**
where you will build your
program

3

Block
Toolbox

1

2

Download your program
to the micro:bit

Hide or Show
Simulator pane

Download

Smiley

Name your Project and Save it
on your computer

Using the micro:bit

Tools you will <u>commonly</u> use to start with:

- Basic
  - Starting command (if click spacebar, tab, etc)
  - Shows LED pad
- Input
  - Uses the inputs to collect info
- LED
  - Using specific LED lights on the screen
- Loops
  - Do you want things to happen once, twice, forever
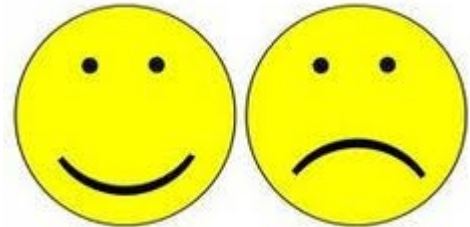- Logic
  - Conditional algorithms (if … then …)
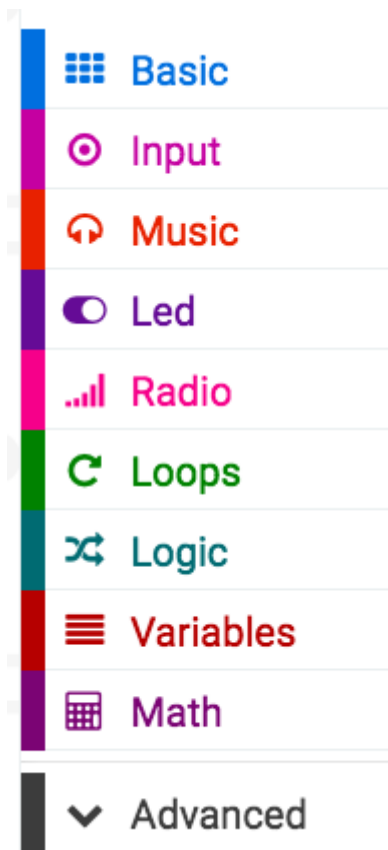
Using the micro:bit - Happy face, Sad face

1) Write out what you want the micro:bit to do in basic terms
   a) Blank screen
   b) Press button A to get a happy face
   c) Press button B to get a sad face

2) Use the tools to "solve the problem"
   a) What command do you need to start with?
   b) How can you make the faces with the LED lights?

Using the micro:bit - Happy face, Sad face

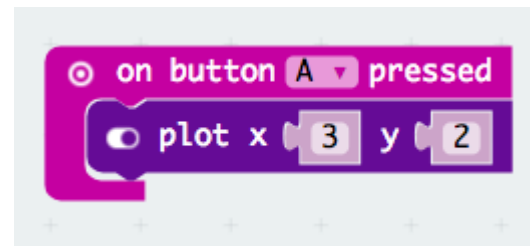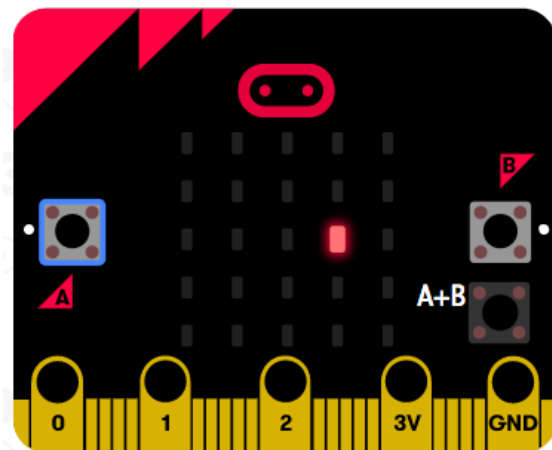The LED screen is a table format, similar to a multiplication table.

0 1 2 3 4

0 1 2 3 4

0 1 2 3 4

0 1 2 3 4

You need to enter the correct grid sequence.

Using the micro:bit - Happy face, Sad face

The LED screen is a table format, similar to a multiplication table.

0 1 2 3 4

0 1 2 3 4

0 1 2 3 4

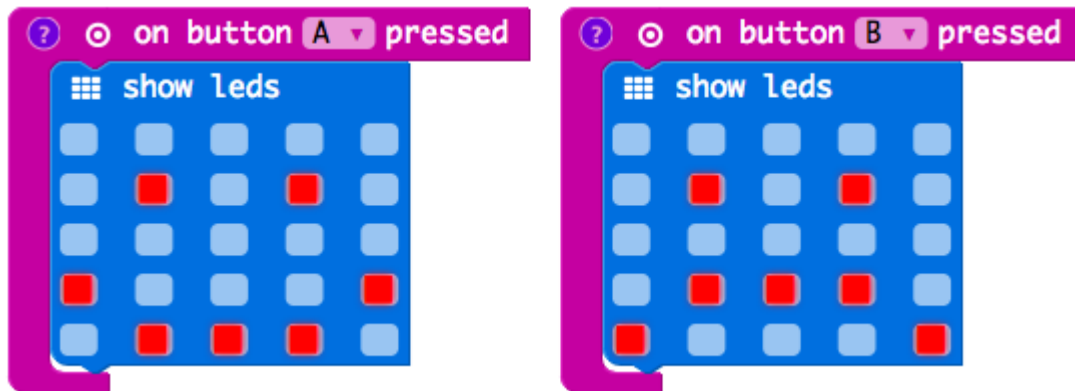0 1 2 3 4

0 1 2 3 4

Now complete for the happy faces!

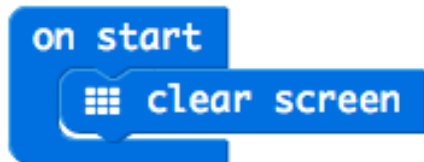What are the plotted points needed for this example?

Using the micro:bit - Happy face, Sad
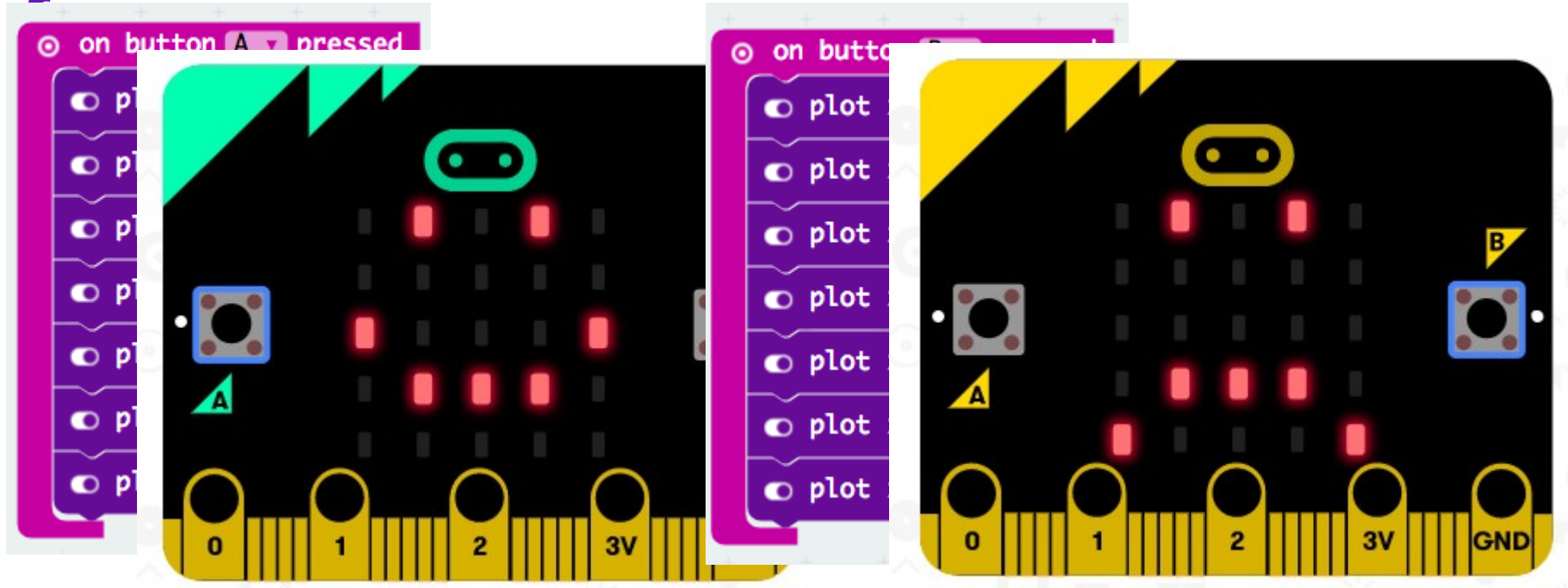
Using the micro:bit - Happy face, Sad face

To make it easier to view the LEDs, you could also use the following:


If you want the screen cleared, you have to include that in your code:

Loading the info to the micro:bit

Previewing in the simulator lets you see what happens before you download.

When the simulator shows your desired output, SAVE your work.

To get the programme on the micro:bit:

- Plug the micro:bit in using the USB
- Attach the battery pack to the micro:bit
- Click "Download"

USB end
for
computer

Battery pack end
for micro:bit

End for
micro:bit
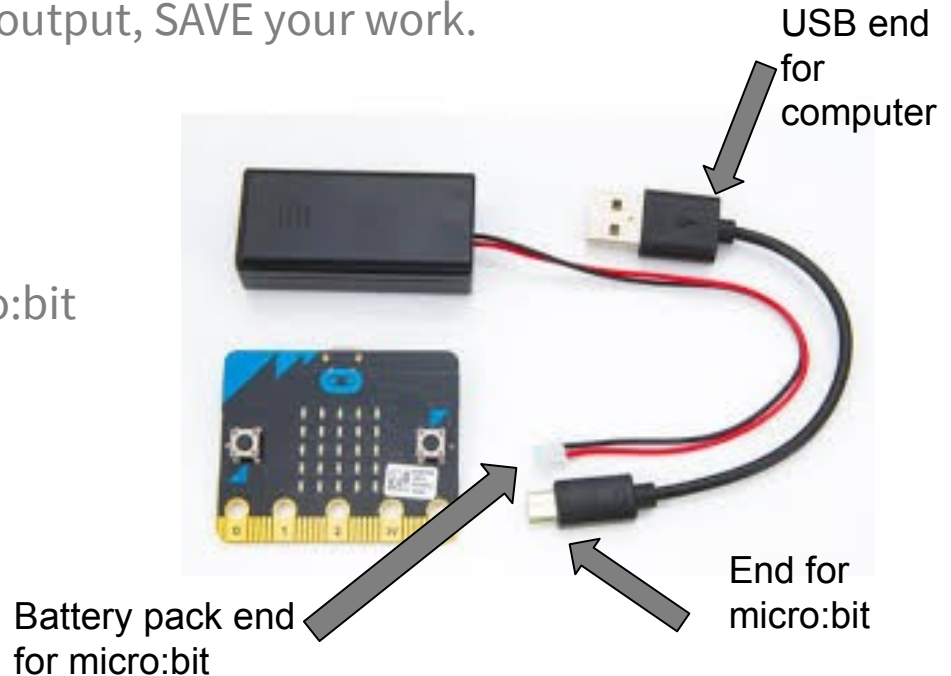
Give it a go...

Code your Micro:bit so that it reads out your name when you click on the "A" button and your partner's name when you click the "B" button.

For something harder (using variables):

Aim to use the "shake" feature, when it randomly selects either of your names when it is shaken.

Computer Science: Use of variables

Similar to when working through a science experiment you have variables.

- Constant: something that does not change
- Variables: something that does change

Examples:

The weather goes from sunny to rainy.

There are 24 hours in a day.

The value of pi is 3.14.

Types of variables

Number -- numerical like a temperature or score

Boolean -- true or false

String -- alphanumeric like a password

Sprite -- represents a dot on a screen with two grid point references

Examples of using variables

Play either Rock Paper Scissors.

What variables are involved?

What about Bingo? Higher/Lower?

Using the micro:bit - Rock Paper Scissors

We are going to design a program to record the Rock Paper Scissors score.

What variables do we need?

1.

2.

3.

What should we call the variables?

What values of the variables can we have?

Using the micro:bit - Rock Paper Scissors

We are going to design a program to record the Rock Paper Scissors score.

What variables do we need?

1. How many times Player A wins
2. How many times Player A wins
3. How many times they tie
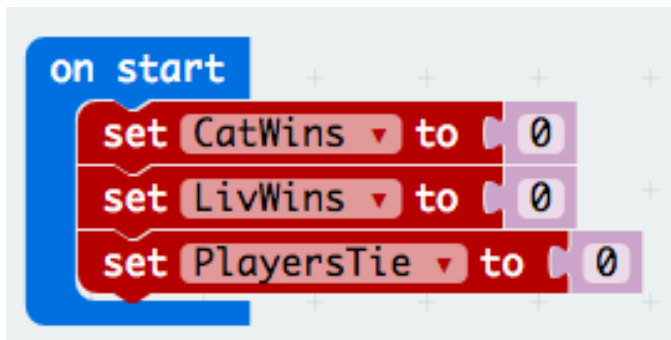
What should we call the variables? PlayerAWins, PlayerBWins, PlayersTie

What values of the variables can we have? 0, 1, 2, 3 … (up to the number of rounds)

Using the micro:bit - Rock Paper Scissors
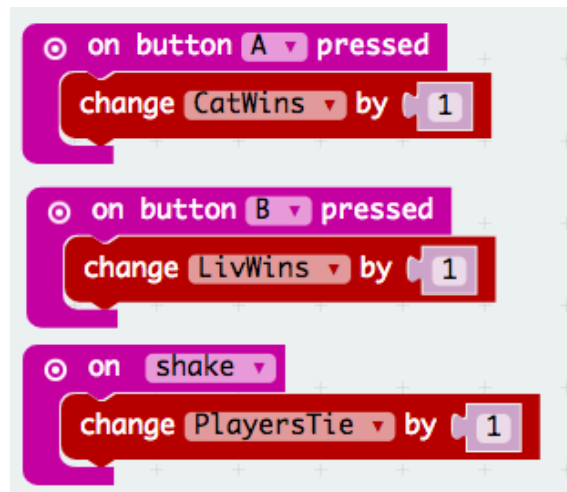
What to consider:

- We are all starting at zero (no one has played yet)
- Each entry will only increase **one** of the variables
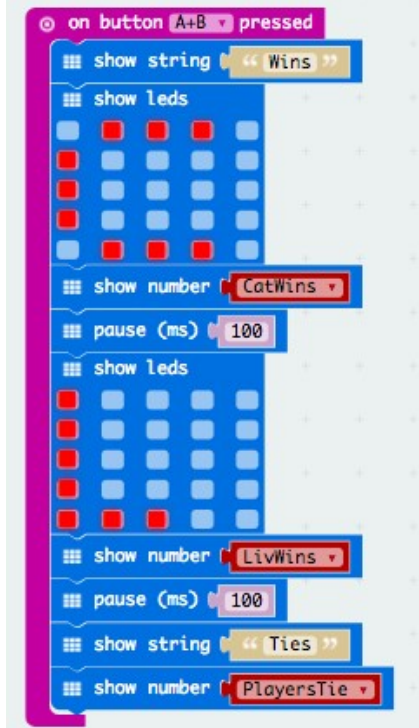   - In maths that's like x+1

Using the micro:bit - Rock Paper Scissors

- Add in an LED screen for each sequence to show the letter for the player's first initial.
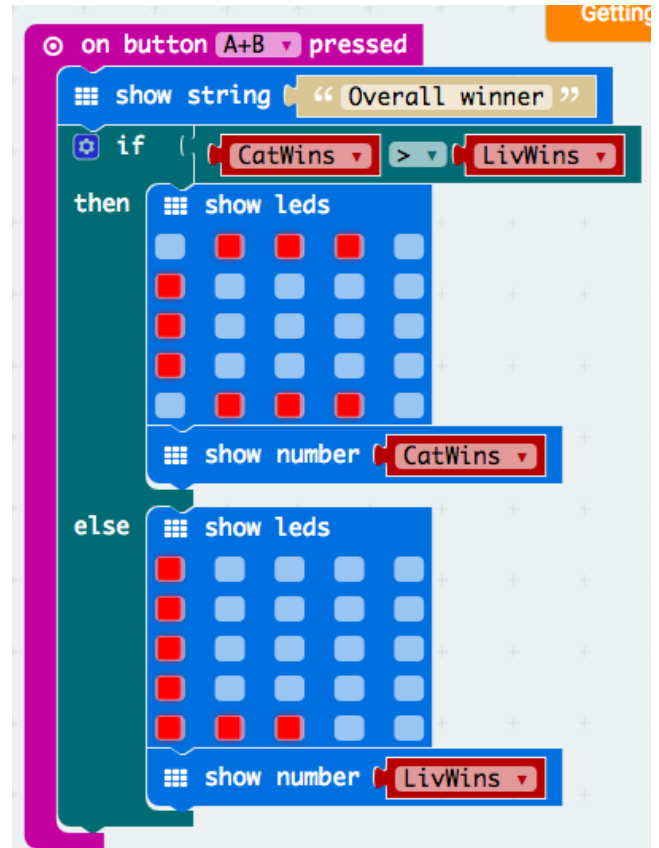- Clear the screen after each variable is displayed
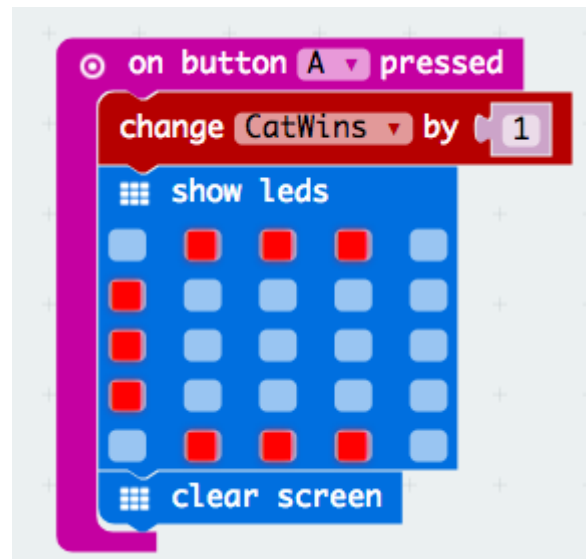
Using the micro:bit - Rock Paper Scissors

To display the overall winner:

MORE ADVANCED ➔

Micro:bot activity

Design something that you can use the micro:bit **to count**. You may need to modify the micro:bit's appearance for this.

Examples:

- Basketball (counts 2 and 3 pointers for each team)
- Pedometer (number of steps)
- Marks of the class (WT, A, M, E)