

Introduction to Programming and Computer Science/Electronics

Lesson 3



Conditionals

Computer programs are instructions telling the computer how to process input and deliver output.

An important part of programming is telling the computer WHEN to perform a certain task. For this, we use something called 'conditionals'. Conditionals get their name because a certain Condition or Rule has to be met.

Conditionals

Have you ever had your parents say the following?

- "If you clean your room, you can go out with your friends."
- "If your homework is done, you can play video games."
- "If you do your chores all week, you get your allowance, else you are grounded."

These are all conditionals! Conditionals follow the format of IF this, THEN that.

IF (condition is met), **THEN** (action performed)

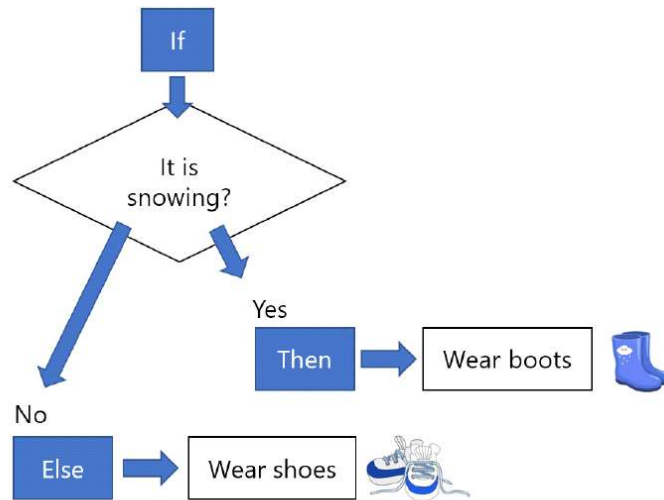
Conditionals

IF (condition is met), **THEN** (action performed), **ELSE** (different action performed)

Example:

- IF it is snowing, THEN wear boots, ELSE wear shoes.

Variables



Keeping Score

Let's Play Rock, Paper, Scissors!

On button A press: choose random number from 0-2
If random number = 0, then display rock icon,
Else if random number = 1, then display paper icon,
Else display scissors icon.

Looping/Iteration

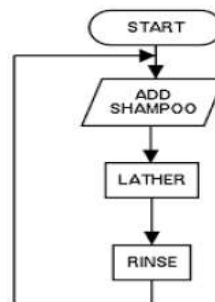
Introduction

In computer programming, iteration is the repetition of a sequence of code. A loop is a form of iteration. A loop repeats code until a certain condition is met.

Looping/Iteration

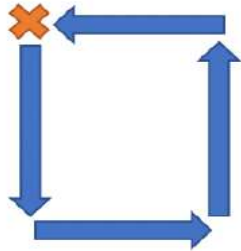
Example:

- 1) Wet hair.
- 2) Apply shampoo to wet hair
- 3) Scrub shampoo into hair
- 4) Rinse shampoo out of hair



Looping/Iteration

- 1) Step forward
- 2) Turn left
- 3) Step forward
- 4) Turn left
- 5) Step forward
- 6) Turn left
- 7) Step forward
- 8) Turn left



Looping/Iteration

Repeat 4 times:

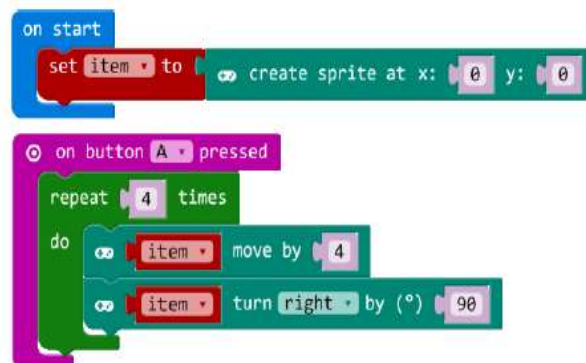
- 1) Step forward
- 2) Turn left

Walk a Square

Code a Sprite to walk a square



Walk a Square



Walk a Square

```
on start
  set item to create sprite at x: 0 y: 0

on button A pressed
  repeat 4 times
    do
      item move by 4
      item turn right by (°) 90
      pause (ms) 100
```

The code is written in Scratch. It starts with an 'on start' block containing a 'set item to' block with a 'create sprite at x: 0 y: 0' block. Below this is an 'on button A pressed' block containing a 'repeat 4 times' loop. Inside the loop is a 'do' block with three sub-blocks: 'item move by 4', 'item turn right by (°) 90', and 'pause (ms) 100'.

Microbit Alarm

```
on shake
  while not button A is pressed
    do
      repeat 2 times
        do
          play tone Middle C for 1/2 beat
          play tone High C for 1/2 beat
```

The code is written in Scratch. It starts with an 'on shake' block containing a 'while not button A is pressed' loop. Inside the loop is a 'do' block with a 'repeat 2 times' loop. Inside the 'repeat' loop is another 'do' block with two 'play tone' blocks: 'play tone Middle C for 1/2 beat' and 'play tone High C for 1/2 beat'.

Project: Get Loopy!

There are many different ways to use the three types of loop blocks.

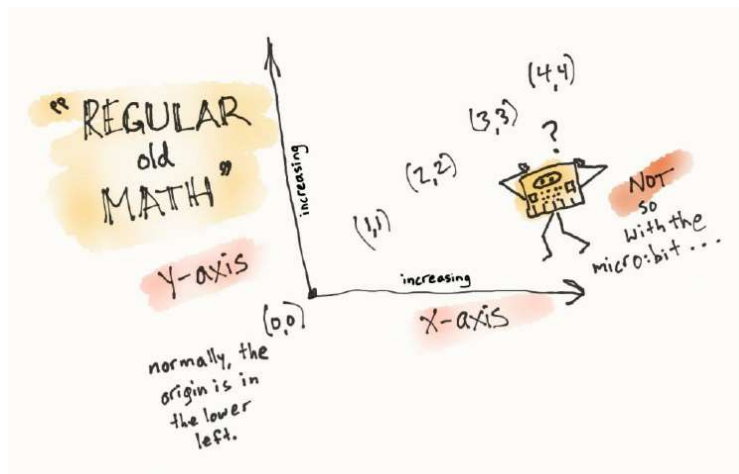
Recall the different common repetitive actions you thought of back at the beginning of this lesson.

- How will you use loops to create something useful, entertaining, or interesting?
- What might you make?

Here are some suggestions:

- Create an animated gif (looping image that changes) and add music that matches.
- Create an animation that repeats for one of the melodies included in Make Code (like Happy Birthday).
- Create different animations that run when different buttons are pressed.
- Create an alarm that includes sound and images. What will set the alarm off? What will make the alarm stop sounding?
- Use servo motors to create a creature that dances and changes its expression while a song plays.

Coordinates



Coordinates

Coordinate grid and JavaScript and the micro:bit

The 5 x 5 grid of LEDs on the micro:bit represent a coordinate grid with a horizontal x-axis and a vertical y-axis. It has an origin and you can refer to the position of the LEDs with coordinate pairs.

It is important however that the students understand the two major differences between the micro:bit LED grid and the coordinate grid that they are used to using in math class:

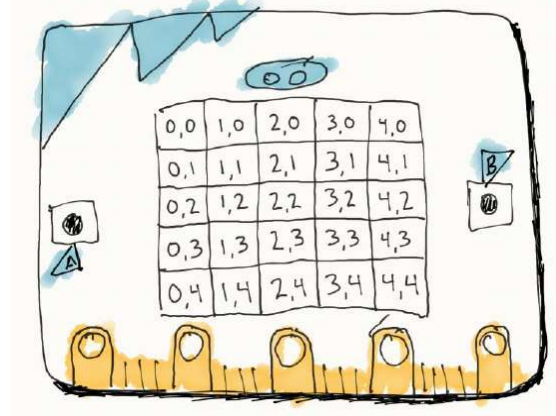
- the origin (0,0) is in the top left corner.
- the values of the y coordinates range from 0 through four and increase from top to bottom.

Note:

- The values of the x coordinates range from 0 through four and increase from left to right just as they do in the coordinate grids used in math class.

Coordinates

MICRO:BIT LED
grid (x,y) coordinates



Travelling Light

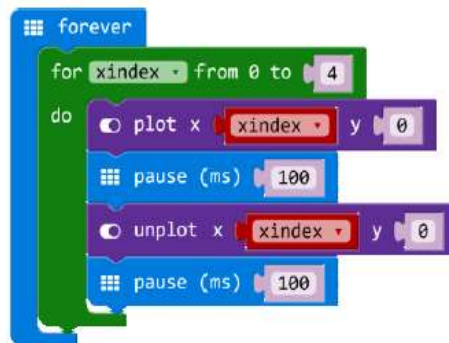
Let's make an led light move across the entire display from left to right, top row to bottom row.

Our pseudocode for the first row might look like this:

```
Turn led x:0, y:0 on
Pause
Turn led x:0, y:0 off
Pause
Turn led x:1, y:0 on
Pause
Turn led x:1, y:0 off
Pause
Turn led x:2, y:0 on
Pause
```

```
Turn led x:2, y:0 off
Pause
Turn led x:3, y:0 on
Pause
Turn led x:3, y:0 off
Pause
Turn led x:4, y:0 on
Pause
Turn led x:4, y:0 off
```

Travelling Light



Travelling Light

Travelling Light along the top row

```
forever
  for xindex from 0 to 4
  do
    plot x xindex y 0
    pause (ms) 100
    unplot x xindex y 0
    pause (ms) 100
```

Travelling Light

Travelling Light along the 5 rows

```
forever
  for yindex from 0 to 4
  do
    for xindex from 0 to 4
    do
      plot x xindex y yindex
      pause (ms) 100
      unplot x xindex y yindex
      pause (ms) 100
```