

June 2020



Programming 2.0

Radio – Week 5

Guru Anginthayya

Agenda – Week 5

- + Review
- + Home work discussion
- + Sensors – Temperature
- + Sensors – Compass
- + Musical Micro:bit
- + Review
- + Homework – 4



Review – Week 4

- + Sensors – Temperature
- + Sensors – Compass
 - + Compass project
- + Algorithms

```
function ShowCompass reading
  if reading ≤ 45 then
    show arrow North
  else if reading ≥ 45 and reading < 90 then
    show arrow North East
  else if reading ≥ 90 and reading < 135 then
    show arrow East
  else if reading ≥ 135 and reading < 180 then
    show arrow South East
  else if reading ≥ 180 and reading < 225 then
    show arrow South
  else if reading ≥ 225 and reading < 270 then
    show arrow South West
  else if reading ≥ 270 and reading < 315 then
    show arrow West
  else if reading ≥ 315 then
    show arrow North West
  else
    show string "Oh Oh"

forever
  set reading to compass heading (°)
  call ShowCompass reading
```



Micro:bit – Fruity Music Notes

```

on start
  set curNoteIndex to 0
  set musicNotes to array of
    260
    293
    330
    350
    391
    440
    493
    523

on button B pressed
  set curNoteIndex to 0

on pin P1 pressed
  play tone musicNotes get value at curNoteIndex for 1 beat
  if curNoteIndex = length of array musicNotes - 1 then
    set curNoteIndex to 0
  else
    change curNoteIndex by 1

on pin P2 pressed
  if curNoteIndex = 0 then
    set curNoteIndex to length of array musicNotes - 1
  else
    change curNoteIndex by -1
  play tone musicNotes get value at curNoteIndex for 1 beat

on button A pressed
  play tone musicNotes get value at curNoteIndex for 1 beat
  change curNoteIndex by 1
  
```

Note	Frequency
C	261
D	297
E	330
F	350
G	392
A	440
B	493
C (next octave)	524



Algorithms - Example

- + Create three musical phrases:
 - + Each phrase should have 4-6 notes.
 - + Each phrase should have a maximum of 4 unique notes.
 - + Each phrase should have a repeating note.
 - + At least one phrase should be ascending.
 - + At least one phrase should be descending.
- + Write an algorithm for each phrase for someone who cannot read music.
 - + This musical phrase is ascending.
 - + This musical phrase contains six notes.
 - + This musical phrase contains four unique notes.
 - + This musical phrase repeats note c three times.

g	a	b	c	d	e	f	g
1	2		3 4 5		6		



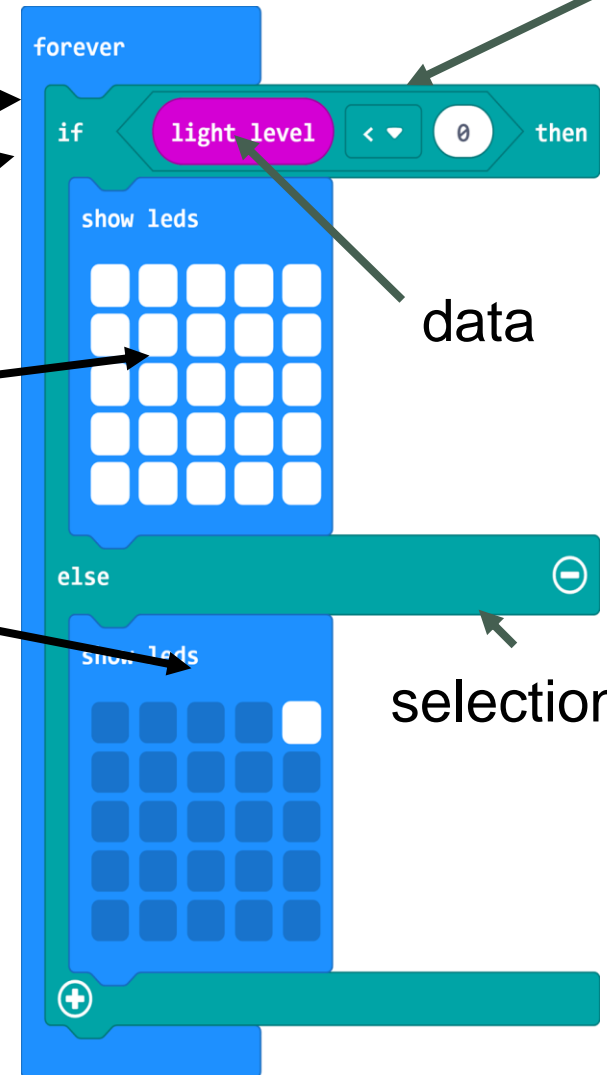
Algorithms and Programs

FOREVER

IF it is dark

THEN turn light on

ELSE turn light off



repetition

→

→

→

→

data

selection



Reading/Optimizing the programs

- + What statements can you make about this program?
- + Which blocks have/haven't we used before?
- + Which computing concepts are being used?
- + How could you improve this program?

```
forever
  if button A is pressed then
    play tone Low A for 1 beat
    play tone Middle C for 1 beat
    play tone Middle D for 1 beat
    play tone Middle D for 1 beat
    play tone Middle E for 1 beat
    play tone Middle E for 1 beat
    play tone Middle E for 1 beat
```

```
forever
  if button A is pressed then
    play tone Low A for 1 beat
    play tone Middle C for 1 beat
    repeat 2 times
      do
        play tone Middle D for 1 beat
    end
    repeat 3 times
      do
        play tone Middle E for 1 beat
      end
    end
```



Flowchart

- + A flowchart is a graphical representations of steps
 - + It was originated from computer science as a tool for representing algorithms and programming logic
 - + Breakdown a process for easier explanation
 - + Help you improve a process

+ Flow Chart Basics

+ Terminator



- + The terminator symbol represents the starting or ending point of the system.

+ Process



- + A box indicates some particular operation.

+ Document



- + This represents a printout, such as a document or a report.

+ Decision



- + A diamond represents a decision or branching point. Lines coming out from the diamond indicates different possible situations, leading to different sub-processes.

+ Data



- + It represents information entering or leaving the system

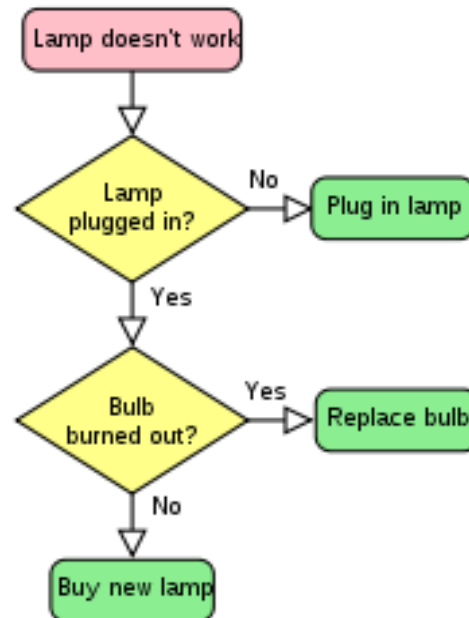
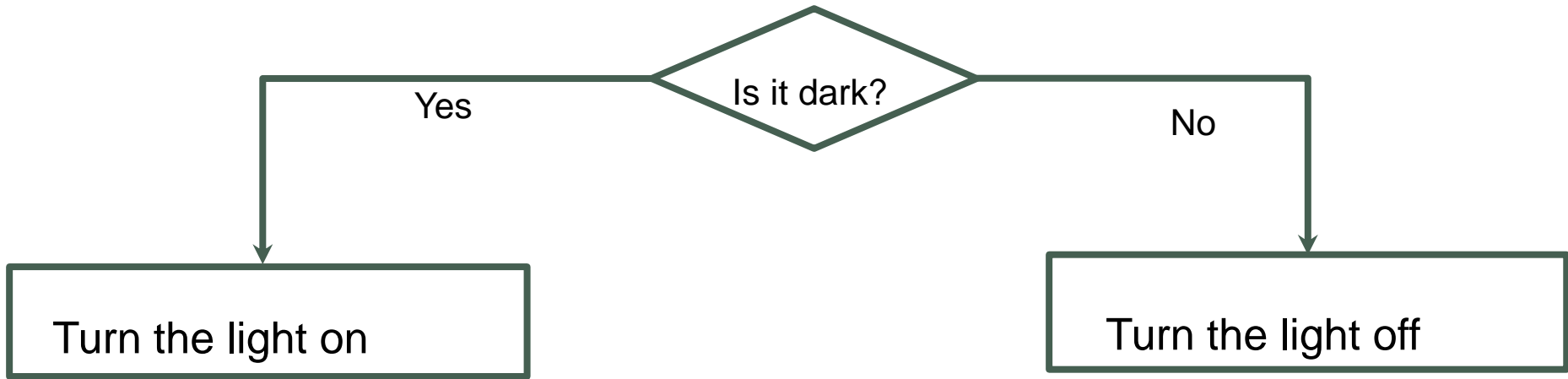
+ Flow



- + Lines represent the flow of the sequence and direction of a process.



Flowchart



Radio

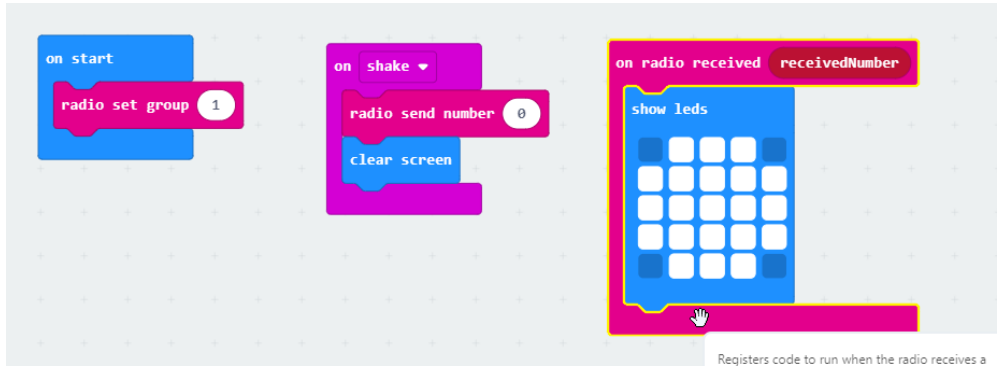
+ Micro:bit – Radio

+ How it works? - <https://www.youtube.com/watch?v=Re3H2ISfQE8&t=10s>



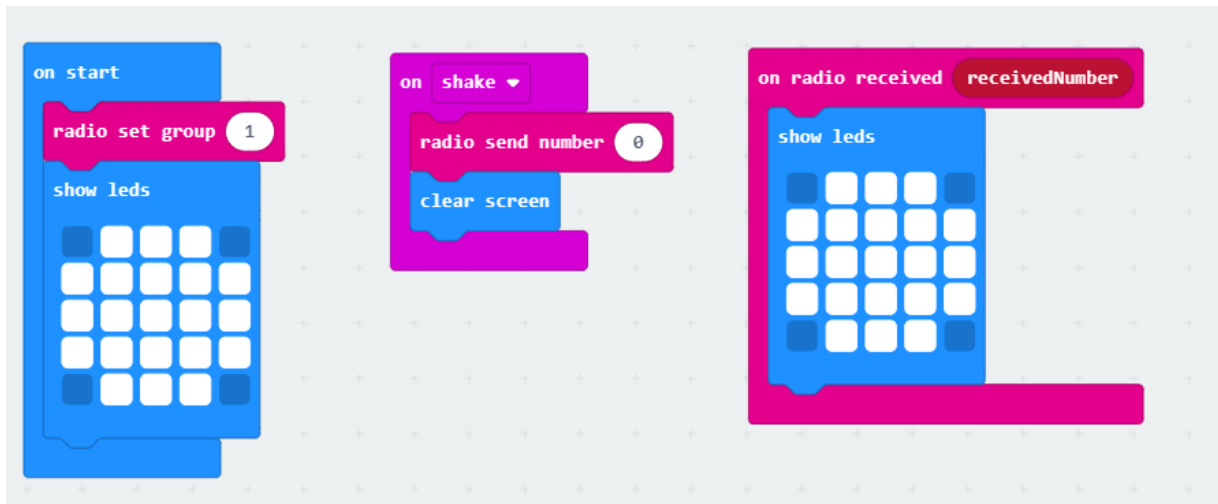
Radio - Pass the Basketball Game

+ Radio 1 – Download



Registers code to run when the radio receives a

+ Radio 2 -

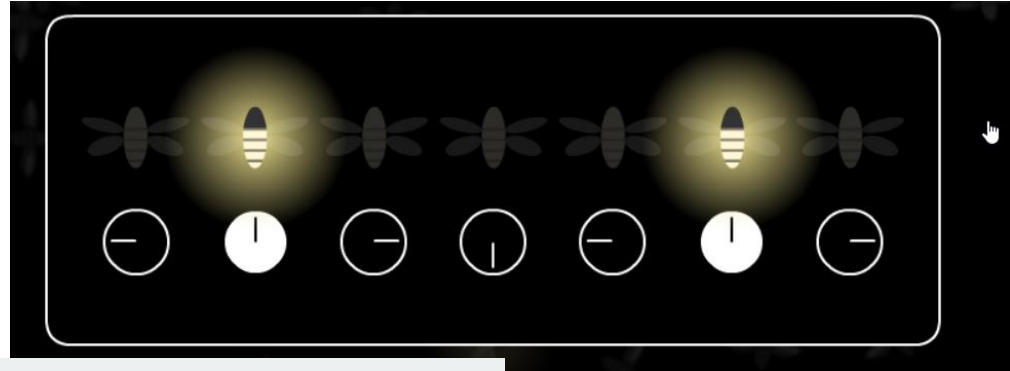


Radio – Firefly

+ How do Fireflies synchronize?

+ <https://ncase.me/fireflies/>

+



```
on start
  radio set group 1
  set clock to 0
  set NoonTime to 8

on radio received receivedNumber
  change clock by 1

forever
  if clock >= NoonTime then
    change score by 1
    radio send number 0
    pause (ms) 200
    set clock to 0
  else
    pause (ms) 100
    change clock by 1
```



Radio – Tell me a secret

The image displays a Scratch script for a radio-based game titled "Tell me a secret". The script is organized into three main sections:

- on start:** A blue block containing a pink "radio set group" block with the value "10".
- on button A pressed:** A pink block containing a pink "radio send string" block with the value "yes", followed by a blue "show icon" block, a blue "pause (ms)" block with the value "500", and a blue "clear screen" block.
- on button B pressed:** A pink block containing a pink "radio send string" block with the value "no", followed by a blue "show icon" block, a blue "pause (ms)" block with the value "500", and a blue "clear screen" block.

The right side of the script features a large pink "on radio received" block with the variable "receivedString". Inside this block, there are two conditional "if" blocks:

- The first "if" block checks if "receivedString" equals "yes". If true, it executes a blue "show icon" block, a blue "pause (ms)" block with the value "500", and a blue "clear screen" block.
- The second "if" block checks if "receivedString" equals "no". If true, it executes a blue "show icon" block, a blue "pause (ms)" block with the value "500", and a blue "clear screen" block.

The script concludes with a long teal block containing a plus sign (+) at the beginning, indicating the end of the code.



References

+ Micro:bit Educational Foundation microbit.org

