

June 2020



Programming 2.0

Music – Week 4

Guru Anginthayya

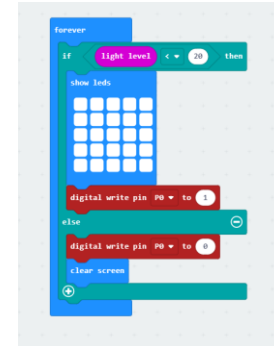
Agenda – Week 3

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

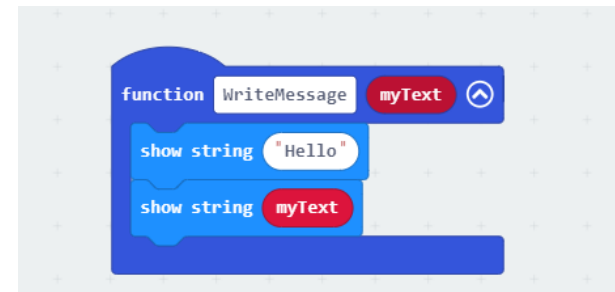


Review – Week 3

- + Light sensor – How light sensor works
 - + Father's day – Night light for your dad
- + Functions
 - + Modules of code that accomplish a specific task
 - + Allow us to reuse code instead of rewriting it
 - + Built-in / System Functions
 - + examples – ShowString



```
forever loop
  if light level <= 20 then
    show leds
  else
    digital write pin P0 to 1
  digital write pin P0 to 1
  clear screen
```



```
function WriteMessage myText
  show string "Hello"
  show string myText
```



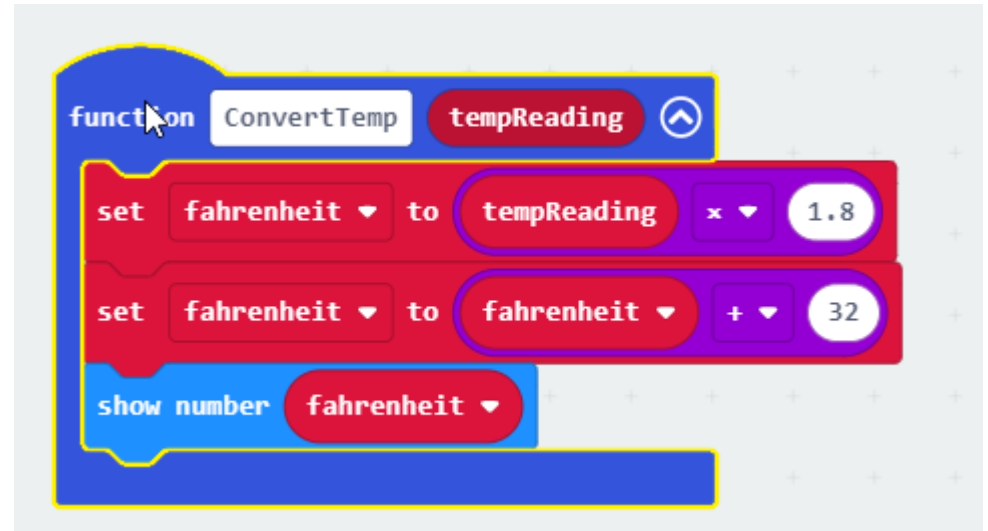
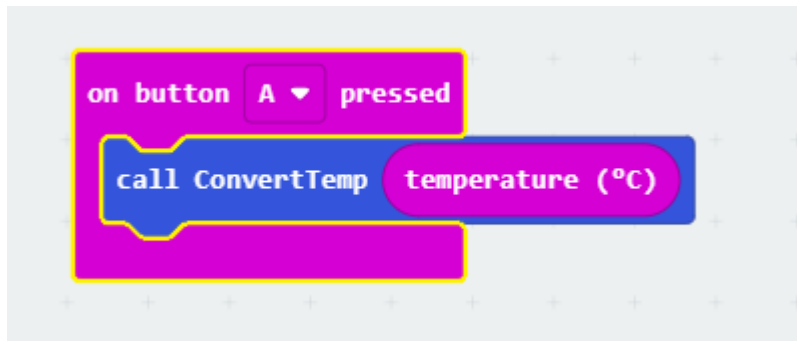
Sensors

+ Temperature sensor

+ How it works?

+ Video - <https://www.youtube.com/watch?v=T4N8O9xsMA>

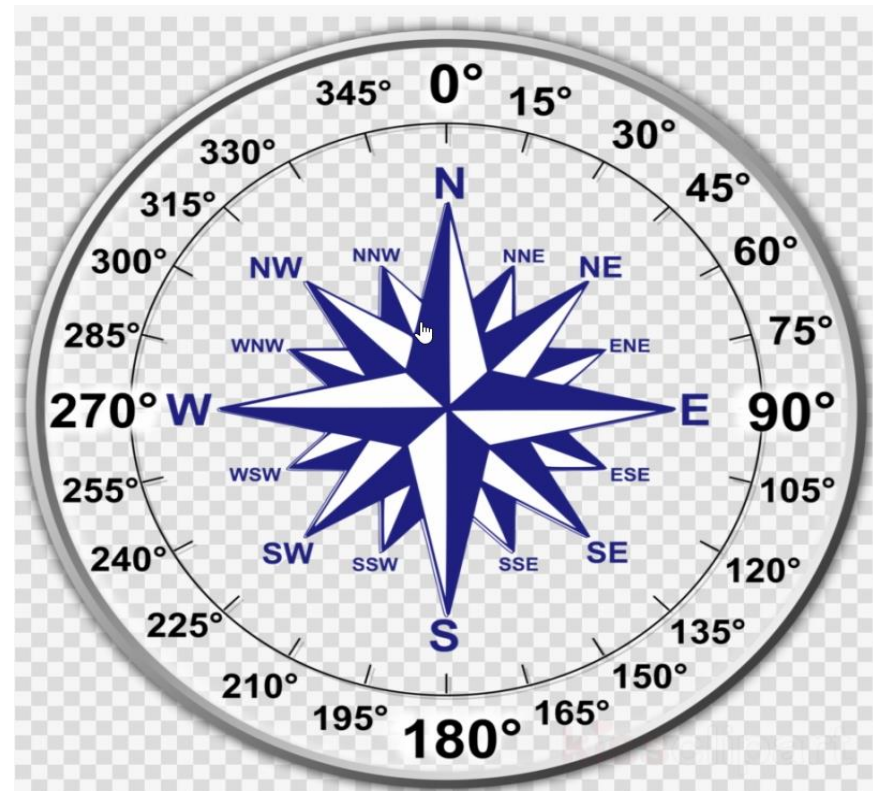
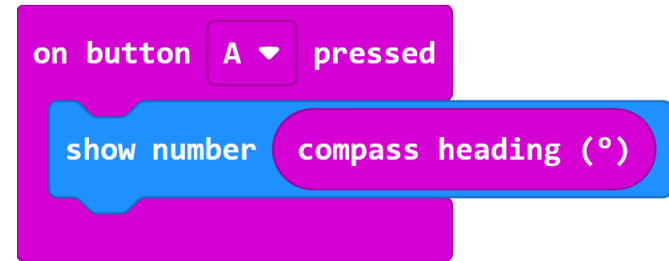
+



Sensors

+ Compass sensor

- + If Reading < 45 \rightarrow N Or > 315
- + If Reading between 45 & 135 \rightarrow E
- + If Reading between 135 & 225 \rightarrow S
- + If Reading $<$ between 225 & 315 \rightarrow W



Algorithm

- + An algorithm is a fancy to-do list for a computer.
- + Algorithms take in zero or more inputs and give back one or more outputs.
- + A recipe is a good example of an algorithm because it tells you what you need to do step by step. It takes inputs (ingredients) and produces an output (the completed dish).
- + Sorting algorithm
 1. Pick up all of the cards.
 2. Pick a card from your hand and look at the color of the card.
 3. If there is already a pile of cards of that color, put this card on that pile.
 4. If there is no pile of cards of that color, make a new pile of just this card color.
 5. If there is still a card in your hand, go back to the second step.
 6. If there is not still a card in your hand, then the cards are sorted.
 7. You are done.



Algorithm

- + Clap for four beats, rest for four beats.
- + Create a musical phrase to play in the rest gap.
- + If you are pointed at, play your musical phrase during the rest.

Forever

IF I am pointed at **then**

ELSE



Micro:bit – Fruity Music Notes

```

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

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

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



Musical Micro:bit

- + 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		



Reading 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
    repeat 4 times
      do
        play tone Middle E for 1 beat
```

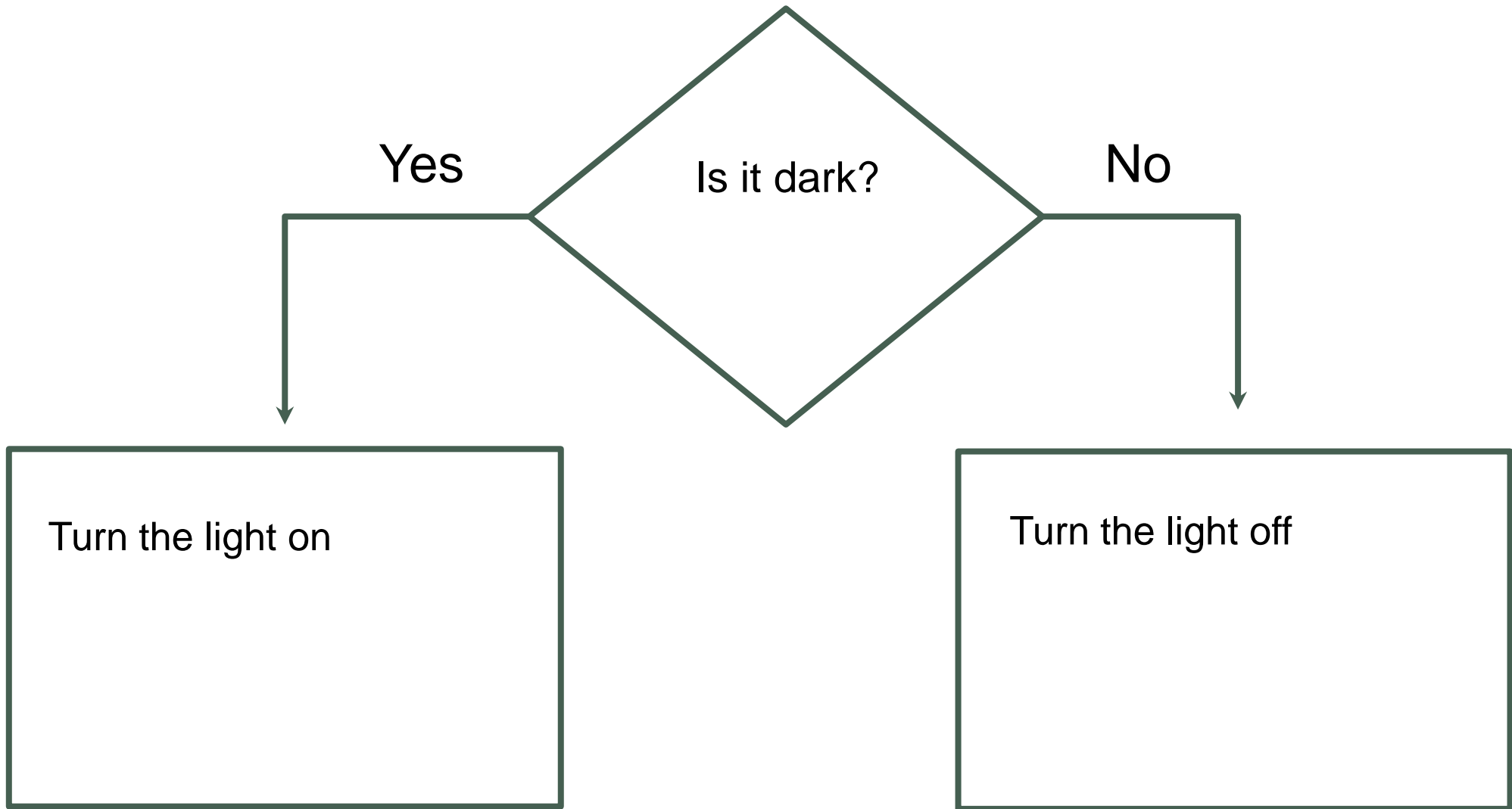


Review

- + Home work – Week 4
 - + Complete the Compass project
 - + Create your own music beats – Play melody
- + Sensors
- + Musical micro:bit
- + How do they collect the data?
- + Sensors



Flowchart



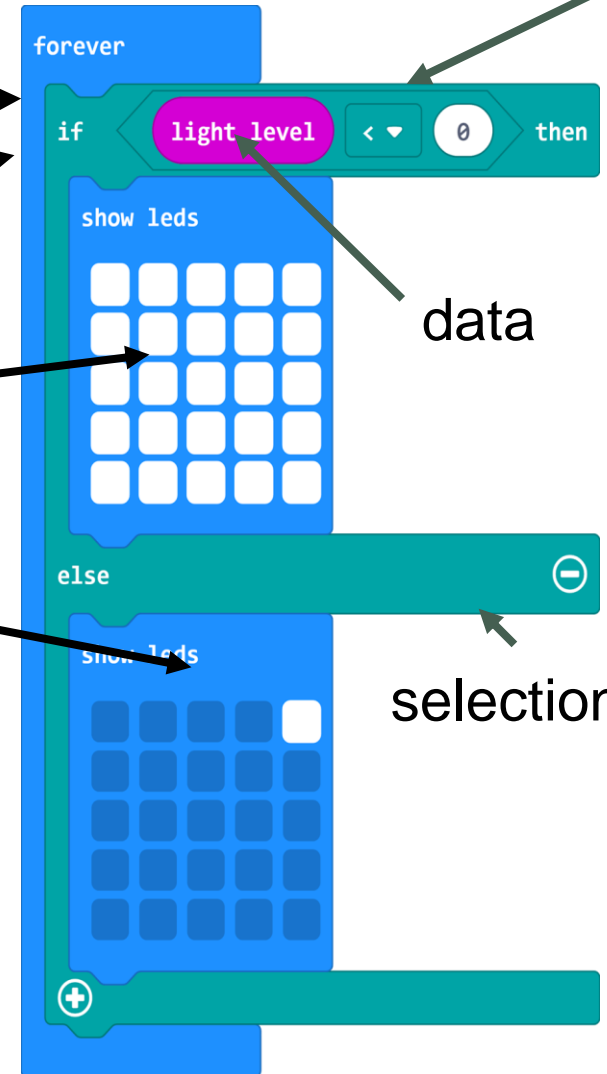
Algorithms and Programs

FOREVER

IF it is dark

THEN turn light on

ELSE turn light off



repetition

data

selection



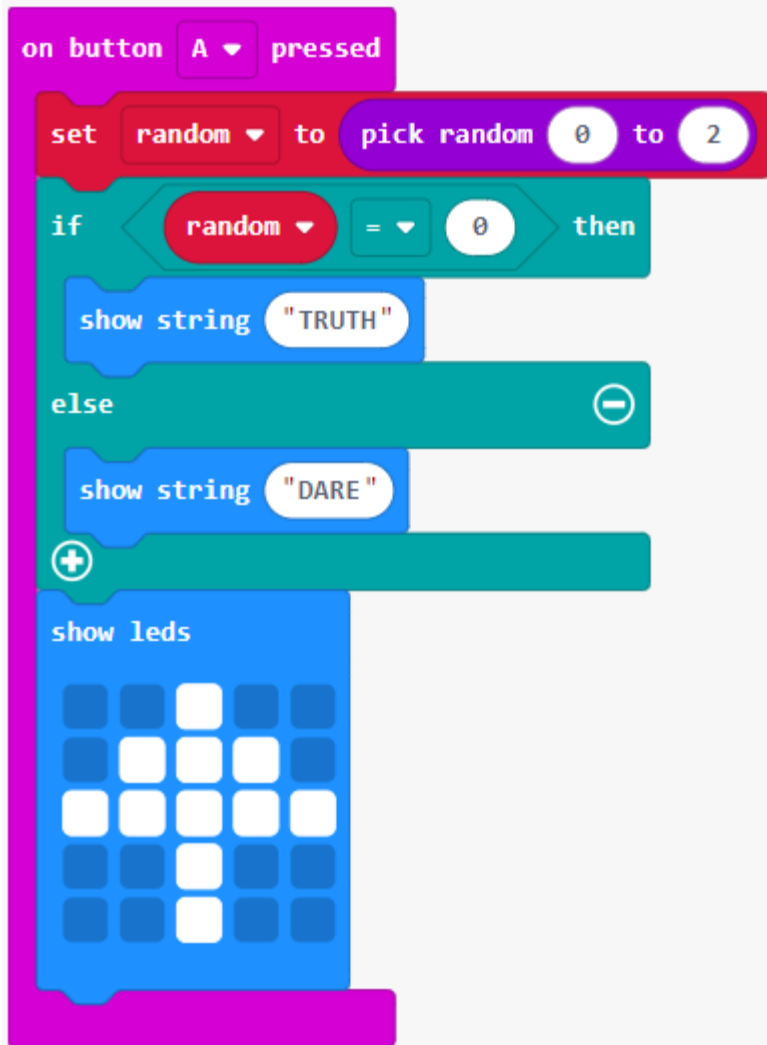
References

+ Micro:bit Educational Foundation microbit.org

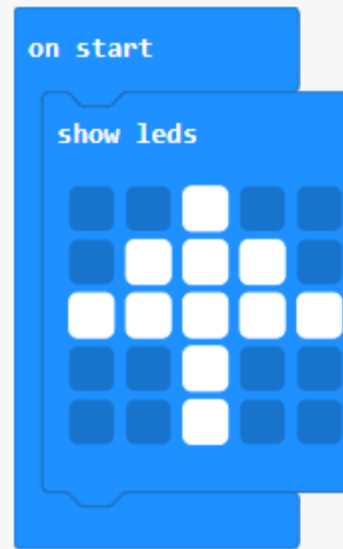


Truth/Dare

```
on button A pressed
  set random to pick random 0 to 2
  if random = 0 then
    show string "TRUTH"
  else
    show string "DARE"
```

A Scratch script starting with an 'on button A pressed' event block. It contains a 'set random to pick random 0 to 2' block, followed by an 'if random = 0 then' conditional block. The 'then' branch contains a 'show string "TRUTH"' block, and the 'else' branch contains a 'show string "DARE"' block. The script ends with a 'show leds' block displaying a 4x4 grid of LEDs with a white pattern.

```
on start
  show leds
```

A Scratch script starting with an 'on start' event block. It contains a 'show leds' block displaying a 4x4 grid of LEDs with a white pattern.