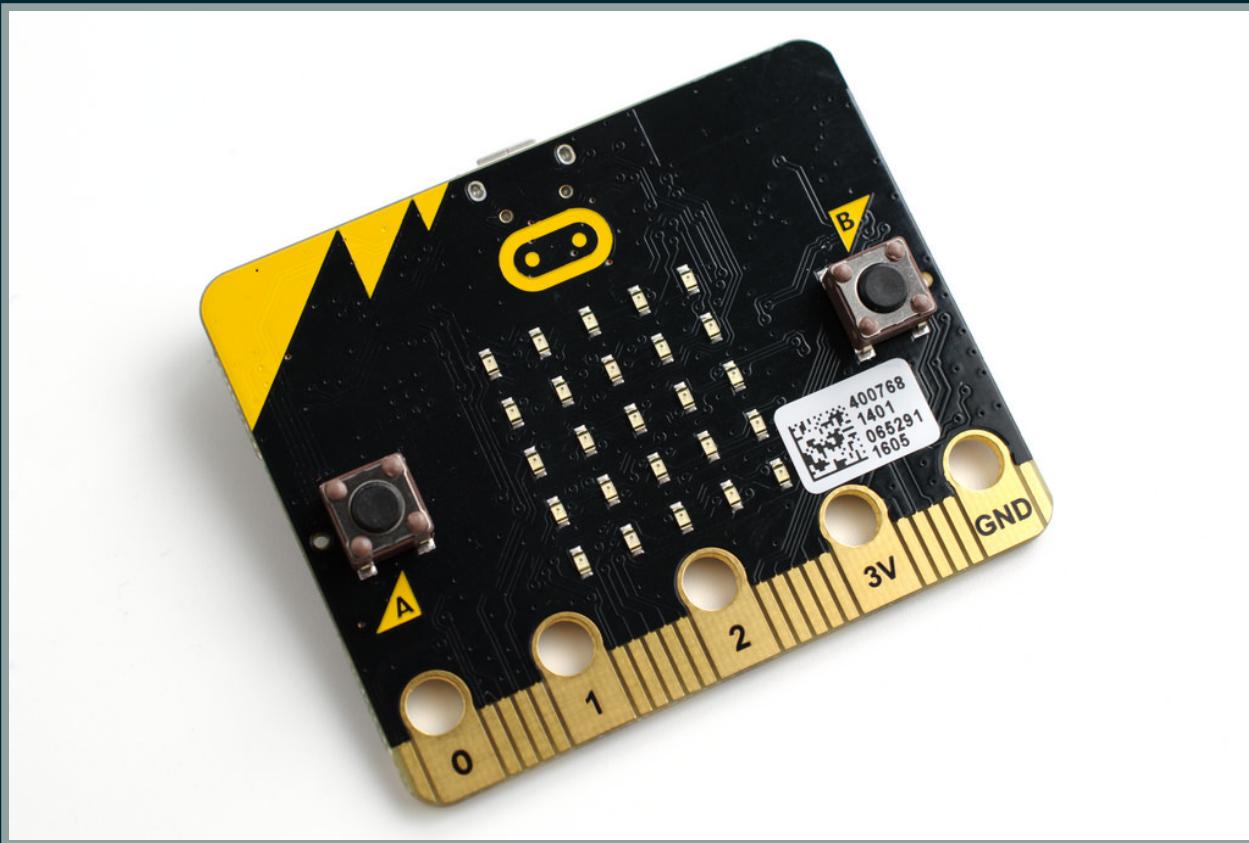


# Micro:Bit

un objet programmable pour faire des maths



# PRÉSENTATIONS

# ANIMATEURS D'ATELIER

QUI ?

Bruno Bourgine & Pascal Padilla

QUOI ?

Professeurs de Mathématiques et Sciences Physiques  
en Lycée Professionnel

OÙ ?

IREM Marseille

# GROUPE INEFLP

Innovation

**FORMES SCOLAIRES INNOVANTES**

Expérimentation

**MICRO-CONTRÔLEUR**

Formation

**ALGORITHME**

# À PROPOS DE CET ATELIER

**PROGRAMMER DES OBJETS CONNECTÉS POUR FAIRE DES MATHS.**

- Comprendre les objets de notre environnement.
- Des objets pour mesurer et communiquer.
- Des objets pour faire des maths.

# DÉROULEMENT DE CET ATELIER (THÉORIQUE)

# DÉROULEMENT DE CET ATELIER (THÉORIQUE)

## 1. Découverte du Micro:bit

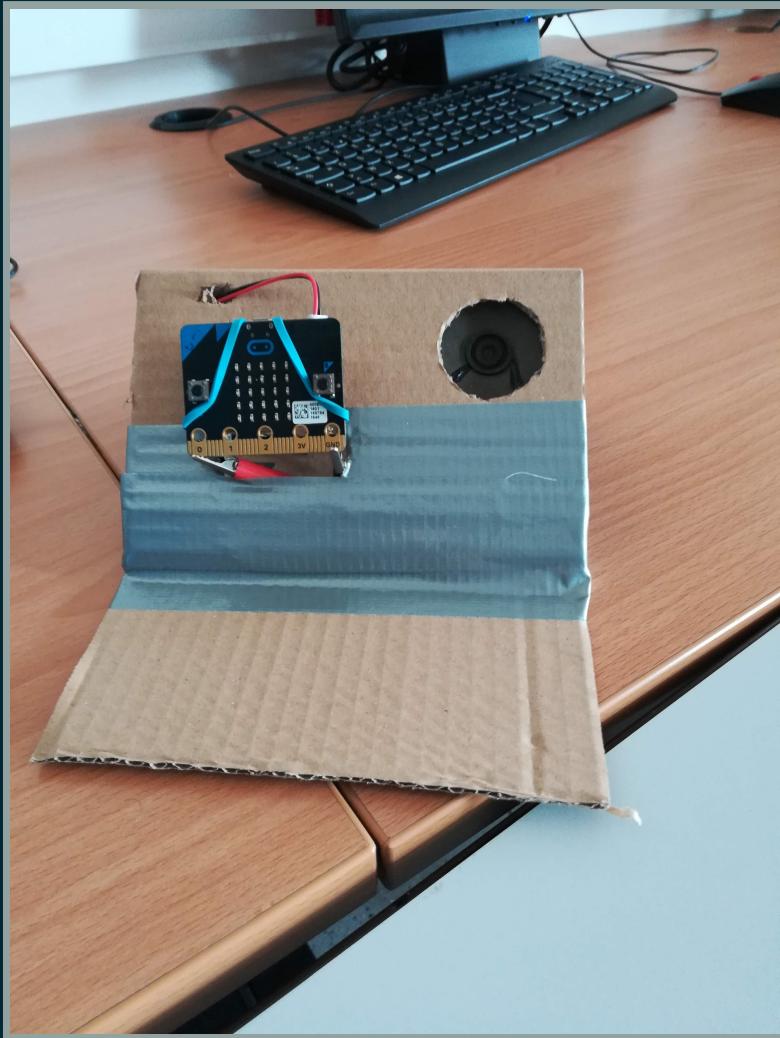
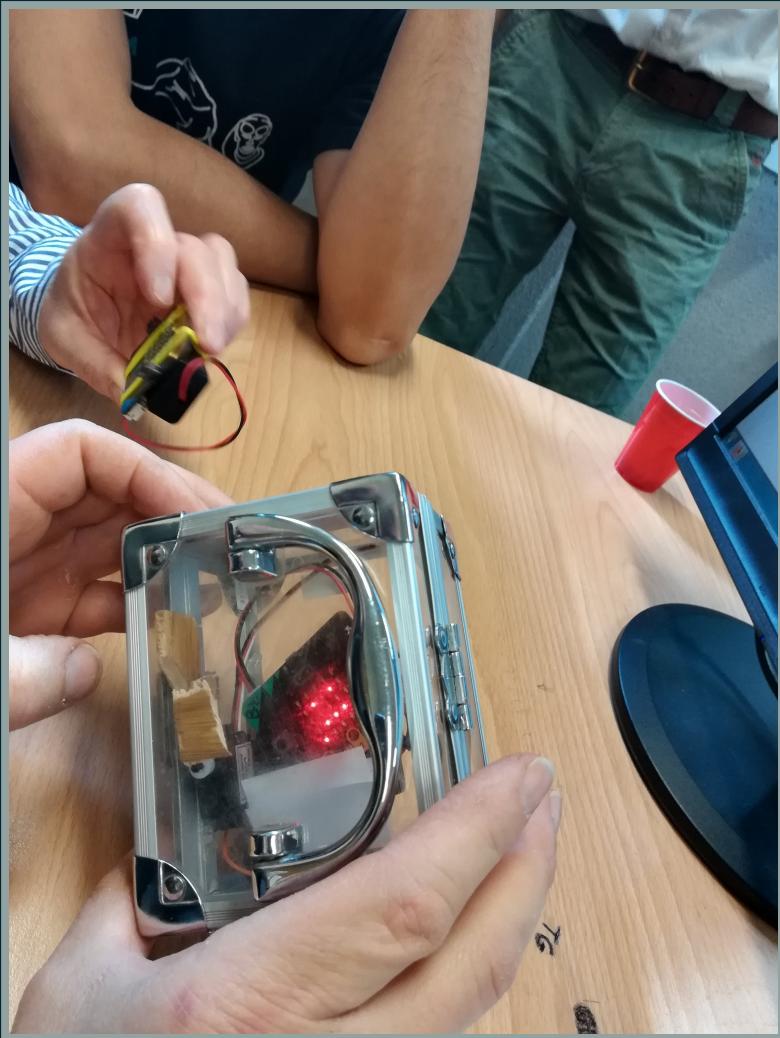
# DÉROULEMENT DE CET ATELIER (THÉORIQUE)

1. Découverte du Micro:bit
2. La programmation par bloc

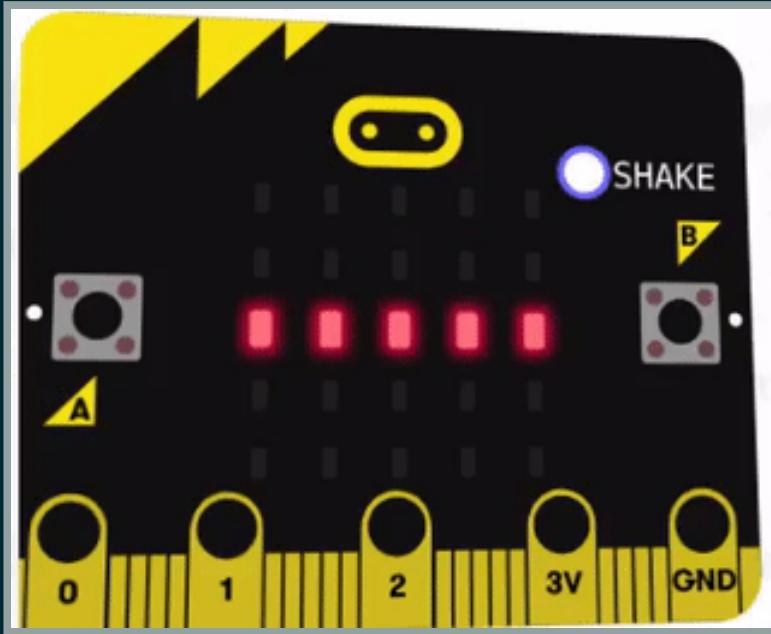
# DÉROULEMENT DE CET ATELIER (THÉORIQUE)

1. Découverte du Micro:bit
2. La programmation par bloc
3. La programmation en Python

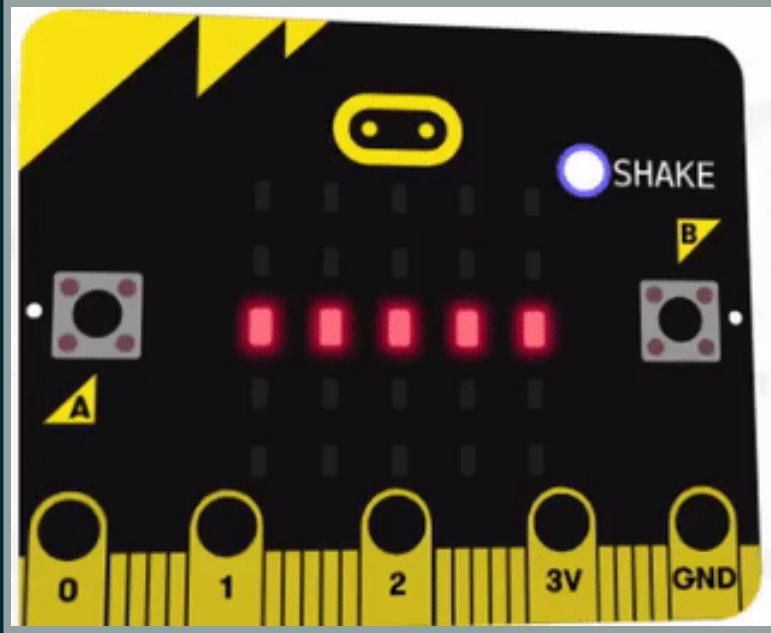
# DÉCOUVRONST LE MICRO:BIT



# ACTIVITÉ DE DÉCOUVERTE

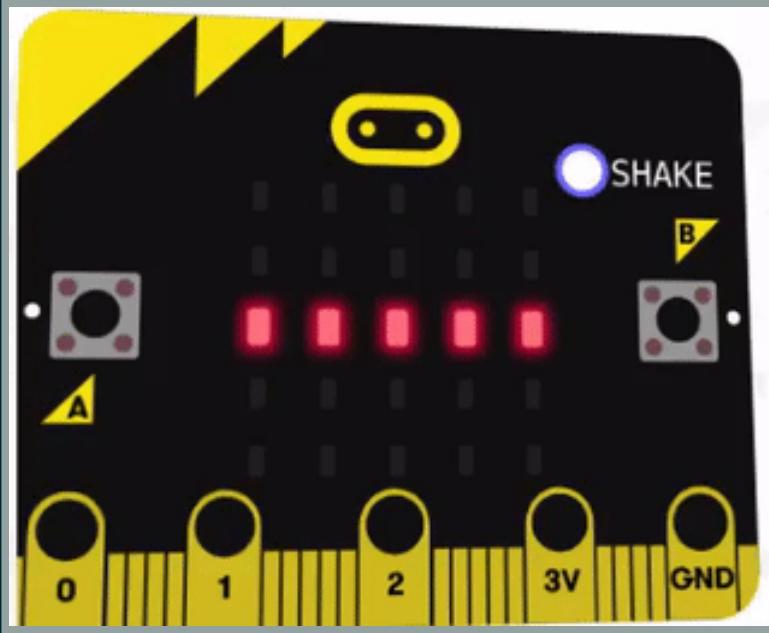


# ACTIVITÉ DE DÉCOUVERTE



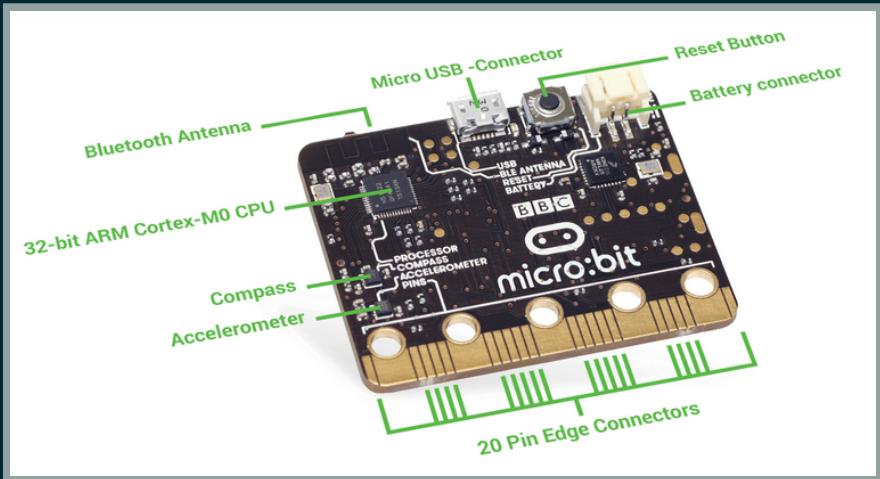
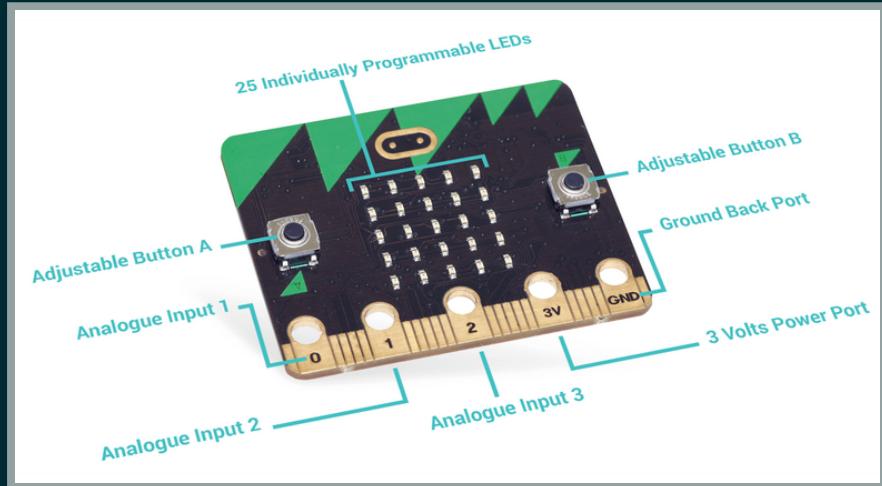
- Mais qu'est-ce que ça fait ?

# ACTIVITÉ DE DÉCOUVERTE



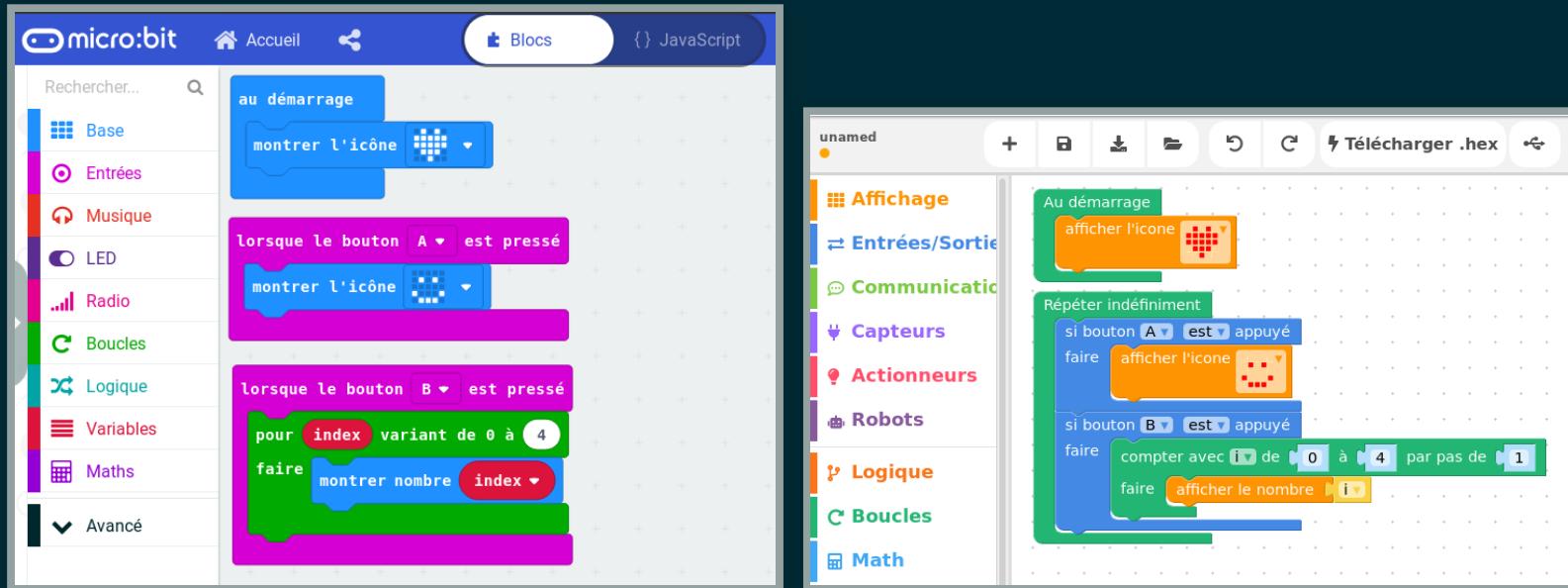
- Mais qu'est-ce que ça fait ?
- Est-ce que c'est équitable ?

# ÉQUIPEMENT DE SÉRIE ...



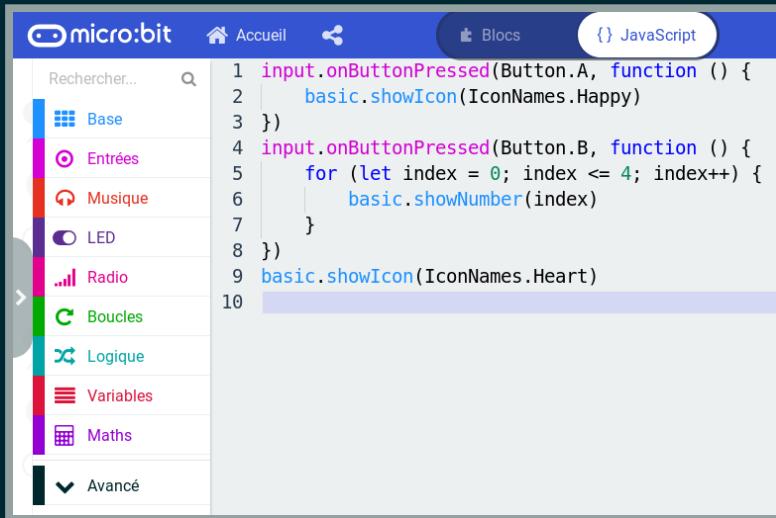
# PROGRAMMATION ...

...AVEC DES BLOCS (MAKECODE, BLOCKLY)



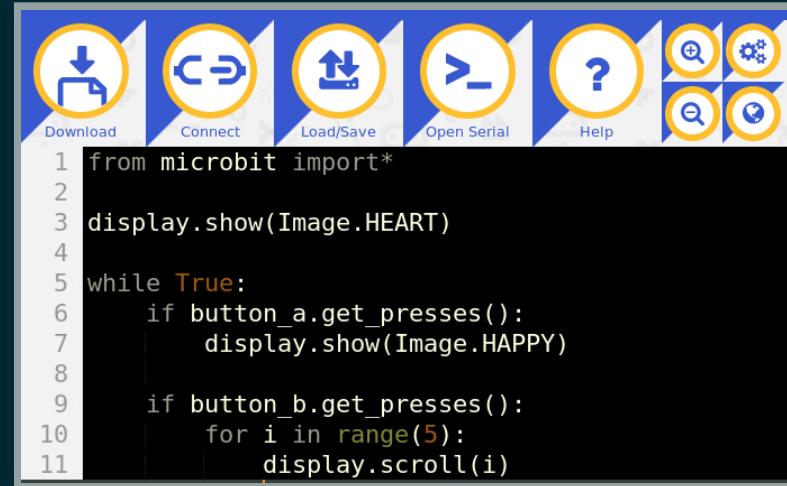
# PROGRAMMATION ...

...AVEC DU CODE (JAVASCRIPT, PYTHON, C/C++)



The screenshot shows the Scratch micro:bit interface. On the left is a sidebar with categories: Base, Entrées, Musique, LED, Radio, Boucles, Logique, Variables, Maths, and Avancé. The 'JavaScript' tab is selected at the top right. The main area contains the following JavaScript code:

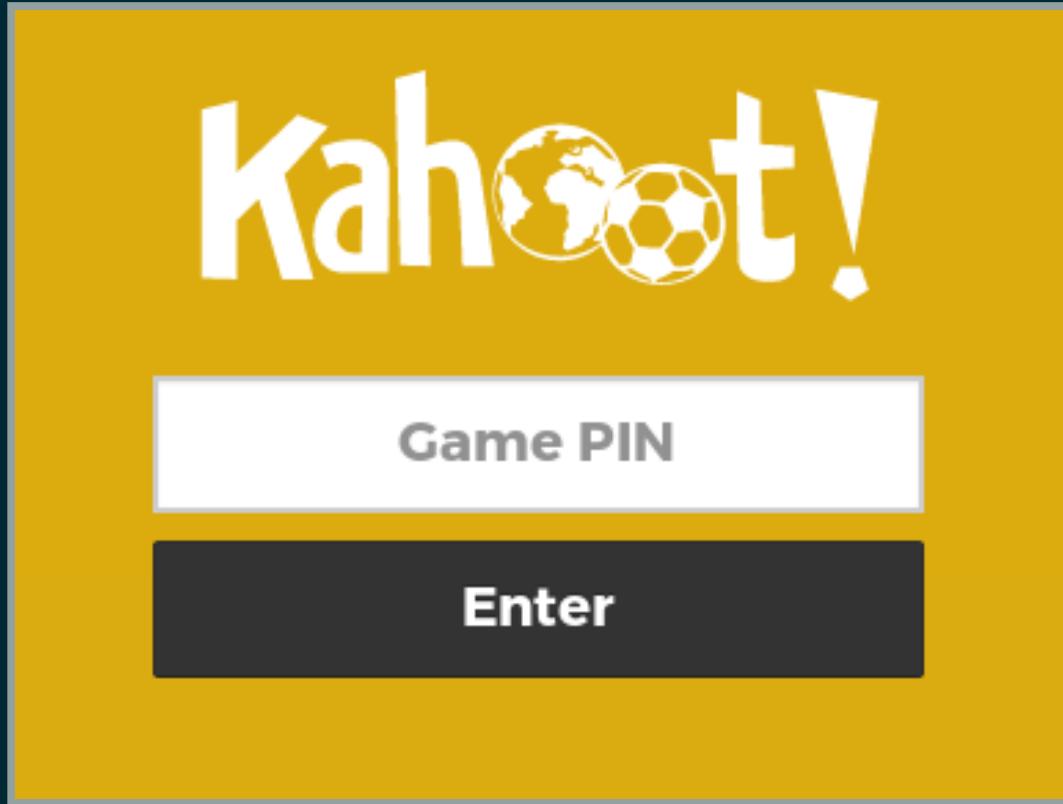
```
1 input.onButtonPressed(Button.A, function () {
2     basic.showIcon(IconNames.Happy)
3 }
4 input.onButtonPressed(Button.B, function () {
5     for (let index = 0; index <= 4; index++) {
6         basic.showNumber(index)
7     }
8 }
9 basic.showIcon(IconNames.Heart)
```



The screenshot shows the Scratch micro:bit interface. At the top, there are five circular icons: Download, Connect, Load/Save, Open Serial, and Help. Below them is a Python script:

```
1 from microbit import*
2
3 display.show(Image.HEART)
4
5 while True:
6     if button_a.get_presses():
7         display.show(Image.HAPPY)
8
9     if button_b.get_presses():
10        for i in range(5):
11            display.scroll(i)
```

# PETIT QUIZZ !



<https://kahoot.it>

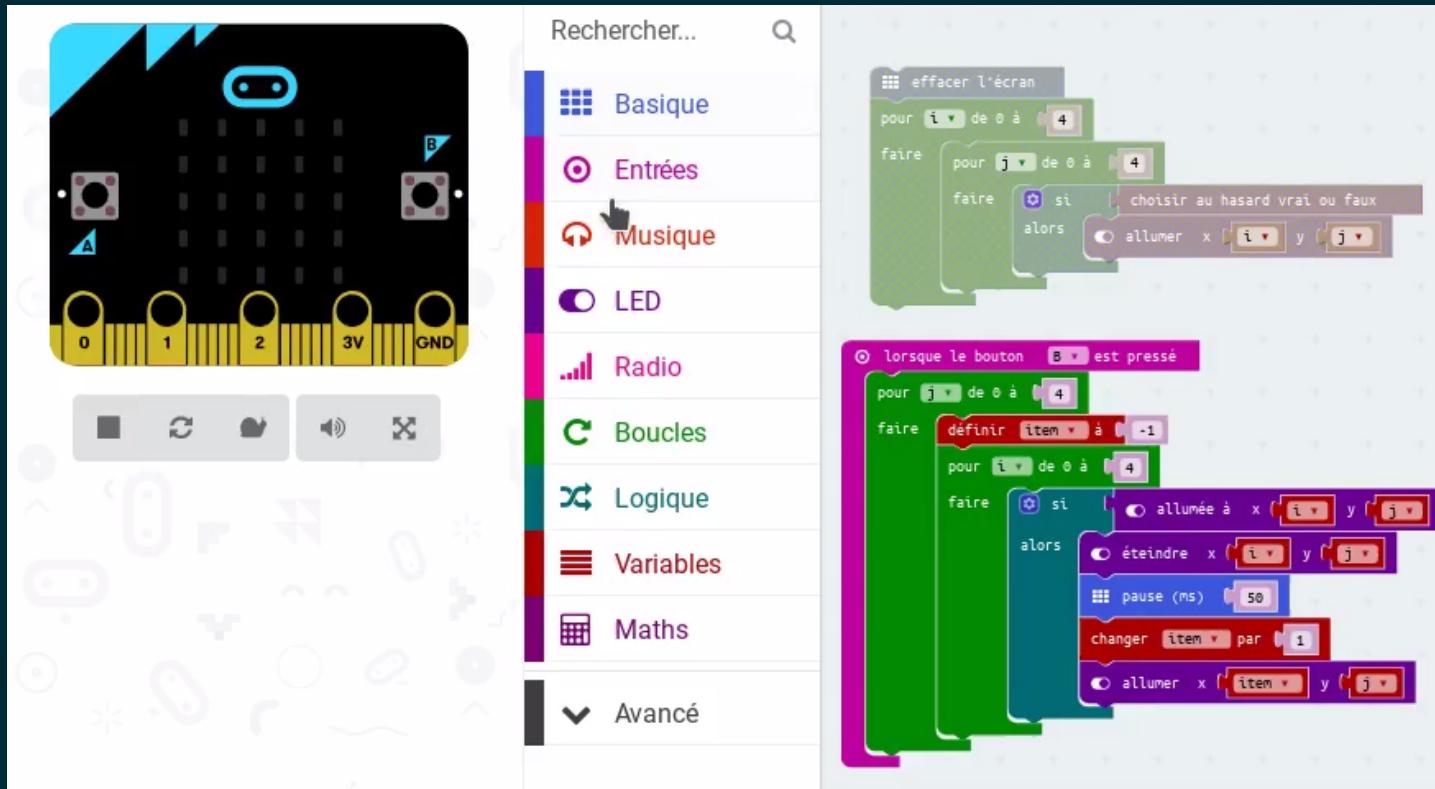
# À PROPOS DU TIRAGE ALÉATOIRE

## DISCUSSION :

- quel intérêt ?
- quelle plus-value ?

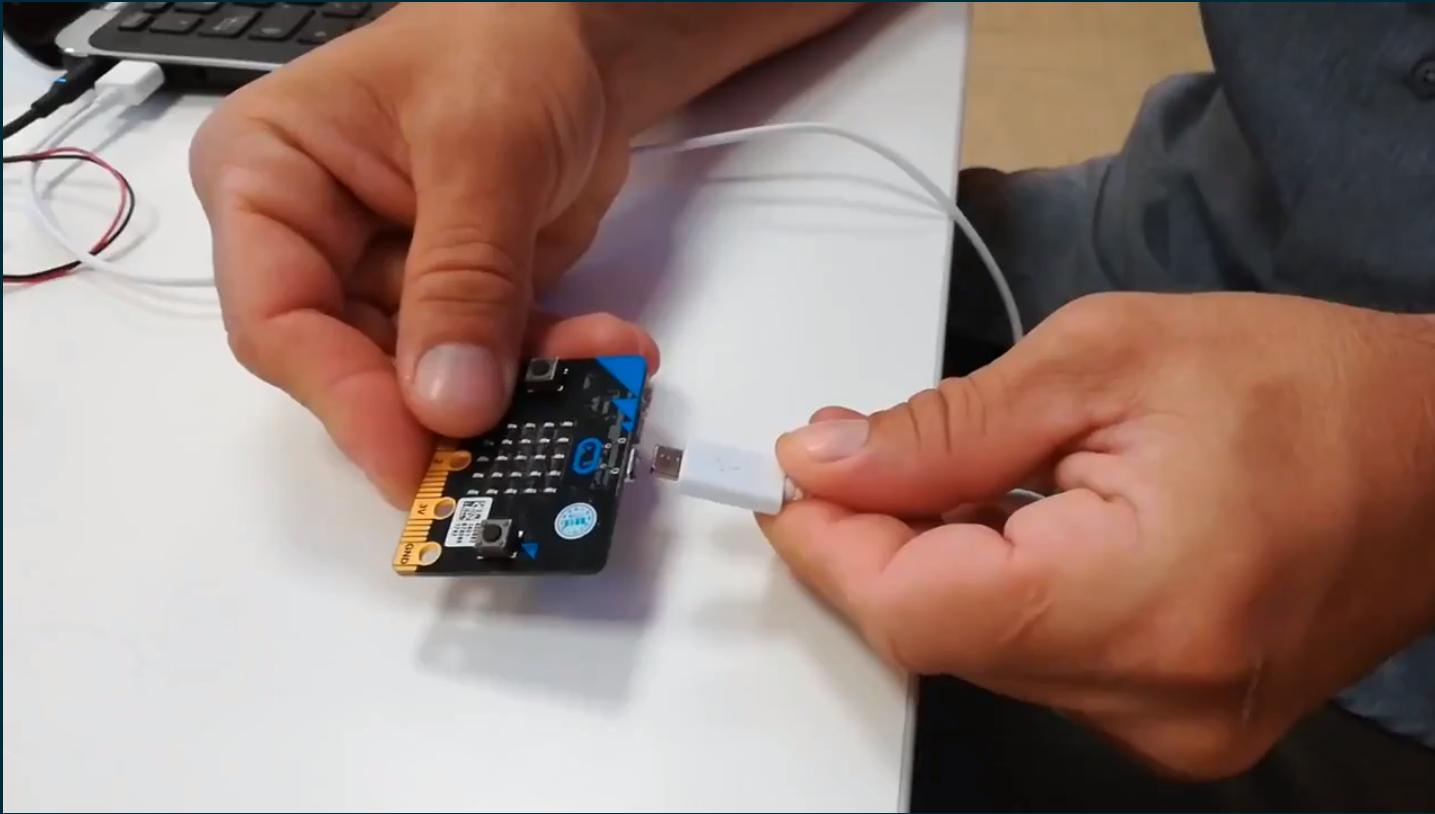
# PROGRAMMATION PAR BLOCS

# MAKECODE



- interface en ligne <https://makecode.microbit.org/>
- programmation par bloc ou en javascript
- simulateur

# MAKECODE



<https://cutt.ly/staosBy>

# EXEMPLES D'ACTIVITÉS ...

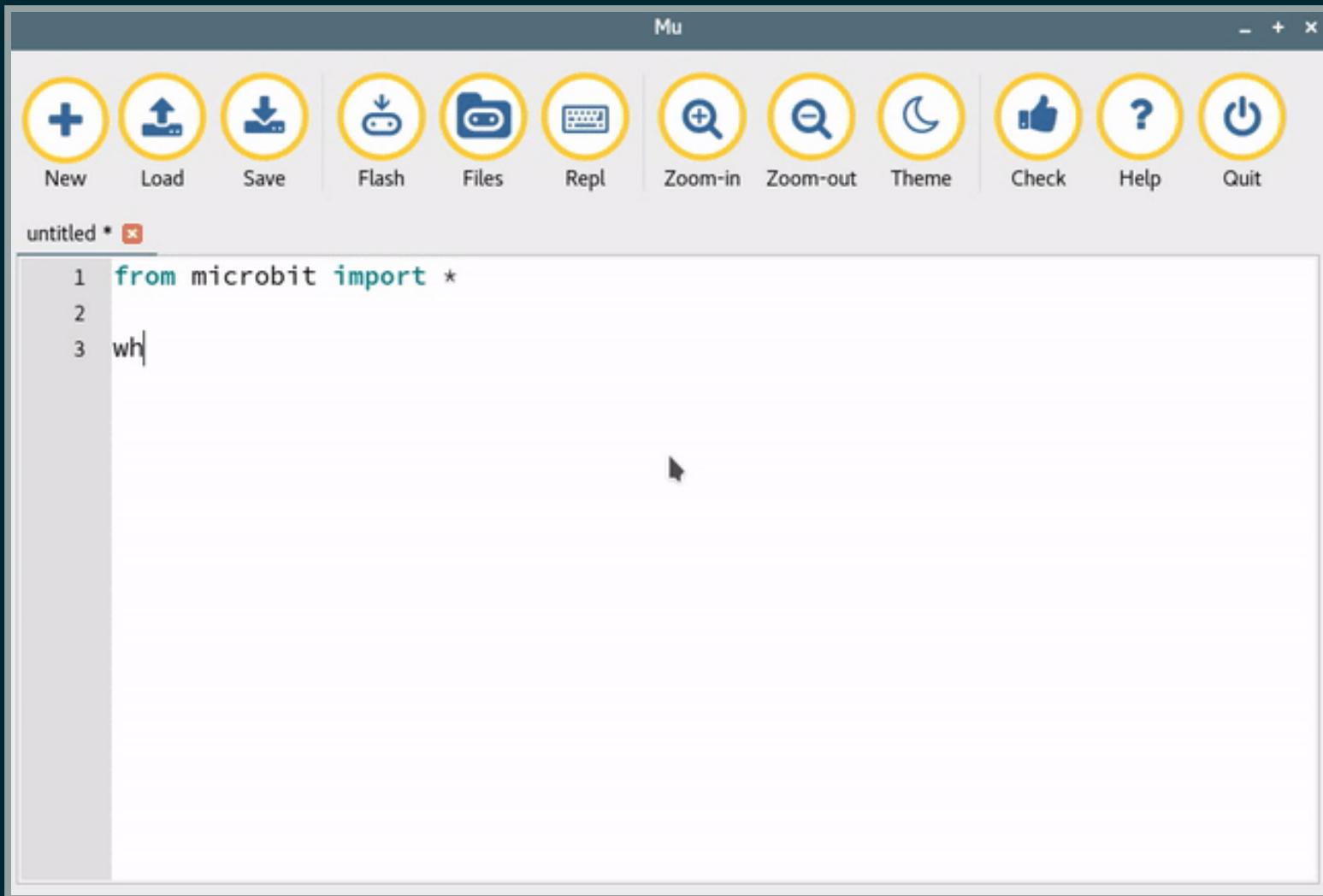
<https://cutt.ly/staosBy>

- Fractions
- Pile ou Face
- Dé
- Fluctuation d'échantillonnage

# PROGRAMMATION

# PYTHON

# MU-EDITOR



<https://cutt.ly/staosBy>

# AFFICHER UN TEXTE

```
from microbit import *
display.scroll("Hello,")
display.show("World!")
```

# TERMINAL ET SORTIE GRAPHIQUE

```
from microbit import *
import random
nb1 = 0
total = 0
for i in range(1000):
    tirage = random.randint(0,1)
    total = total + 1
    nb1 = nb1 + tirage
    nb0 = total - nb1
    print((i, nb1/total, nb0/total))
```

# DES IMAGES

```
from microbit import *
display.show(Image.HAPPY)
sleep(1000)
display.show(Image.ANGRY)
sleep(1000)
display.clear()
```

# LES BOUTONS

```
from microbit import *
sleep(10000)
display.scroll(str(button_a.get_presses()))
```

# LE MOUVEMENT

```
from microbit import *
while True:
    capteur = accelerometer.get_x()
    if capteur > 40:
        display.show(Image.ARROW_E)
    elif capteur < -40:
        display.show(Image.ARROW_W)
    else:
        display.show("-")
```

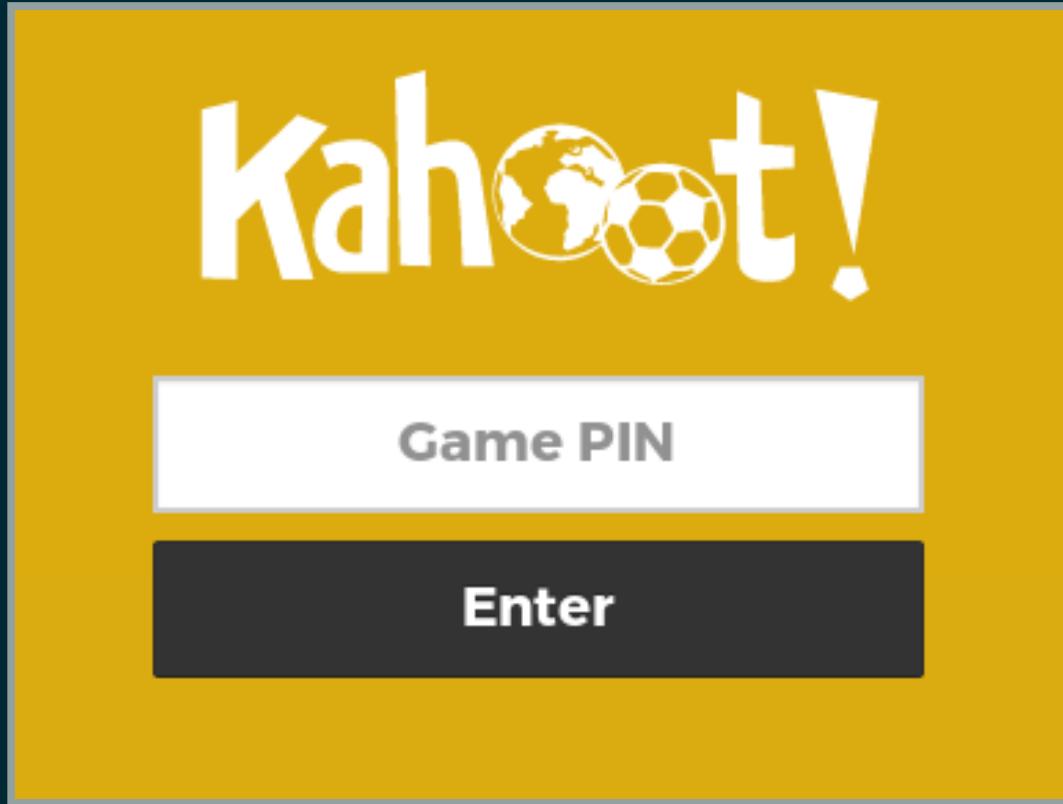
# LES GESTES

```
from microbit import *
import random
button_b.was_pressed()
while True:
    display.show("8")
    if accelerometer.was_gesture("shake"):
        display.clear()
        sleep(1000)
        display.scroll(random.choice(["Oui", "Non"]))
    if button_b.was_pressed():
        display.clear()
        break
```

# LA RADIO

```
from microbit import *
import radio
import random
while True:
    if button_a.was_pressed():
        radio.send("A")
    if button_b.was_pressed():
        radio.send("B")
    # récepteur
    incomming = radio.receive()
    if incomming == "A":
        display.scroll("A")
    if incomming == "B":
        display.scroll("B")
    sleep(20)
```

# PETIT QUIZZ !



<https://kahoot.it>

# POUR ALLER PLUS LOIN

# AUGMENTER VOTRE MICRO:BIT !

- Kit d'extension : ajouter des capteurs et des interfaces

- Scratch : faire interagir micro:bit et Scratch c'est facile!
- Vittascience : passer facilement du bloc à python, et utiliser micro:bit pour des projets scientifiques (ballon solaire)

# CLASSROOM

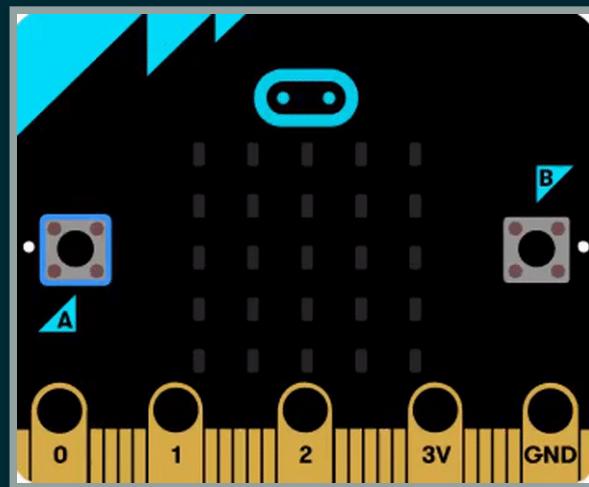
- Partager du code
- Superviser les travaux des élèves
- Sauvegarder des sessions de travail

... Rendez-vous à la  
5ème journée sur l'enseignement de l'informatique  
mercredi 25 mars  
(Luminy)...

# ACTIVITÉ DE RECHERCHE

# PROBLÉMATIQUE

On utilise une boucle de  $n$  itérations pour allumer aléatoirement les diodes du Micro:bit.



# COMBIEN D'ITÉRATIONS SONT NÉCESSAIRES POUR ALLUMER TOUTES LES DIODES ?

# MERCI

# GROUPE INEFLP



# MERCI

## GROUPE INEFLP

### INNOVATION

Formes scolaires innovantes  
**EXPÉRIMENTATION**

Micro-contrôleur  
**FORMATION**

Algorithme

