

ACT 2 Fonctions logiques

COMPETENCES TRAVAILLEES

APP	Mobiliser les concepts et les technologies adaptés au problème (table de vérité)
ANA	Modéliser un problème en utilisant les objets conceptuels de l'informatique pertinents (table de vérité)

Nous allons (re)découvrir les fonctions logiques de base au travers quelques manipulations. On va utiliser pour cela une carte à micro-contrôleur appelée **Micro:bit**.

1 Présentation rapide de la carte utilisée

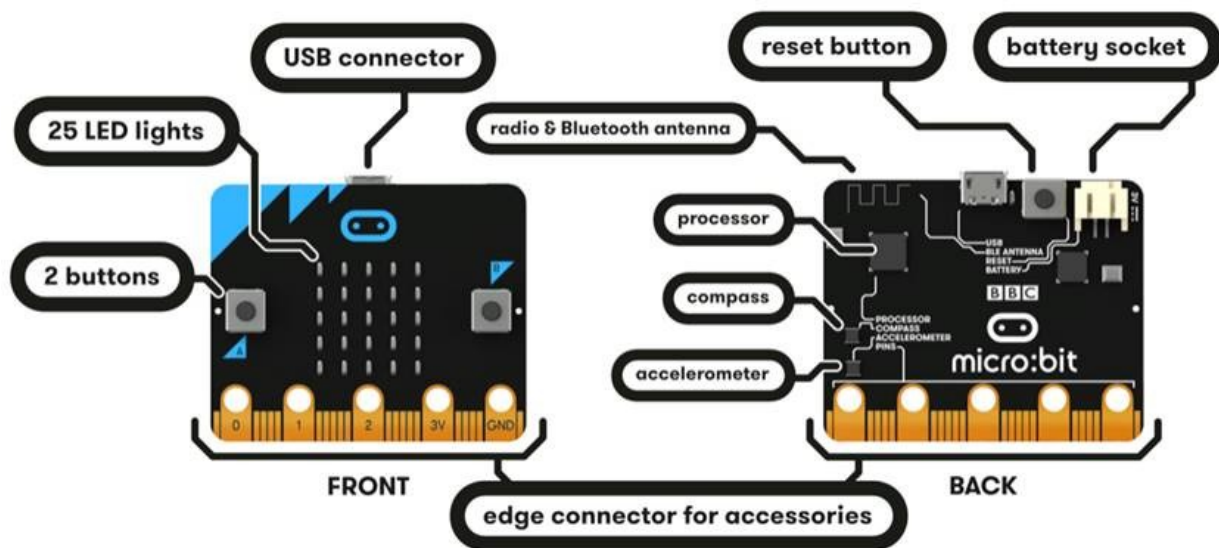
La carte **micro:bit** est un ordinateur à processeur ARM conçu par la BBC pour l'éducation informatique au Royaume-Uni. Cette carte peut se programmer en utilisant plusieurs langages. Nous nous intéresserons ici uniquement à la programmation de la carte sous MicroPython.

Ce petit ordinateur possède la dernière technologie qui équipe les appareils modernes : téléphones mobiles, réfrigérateurs, montres intelligentes, alarmes antivol, robots, etc...

Ainsi, il s'apparente à ce que l'on nomme l'Internet des objets : *Internet of Things*, abrégé IoT.

Un micro:bit est à la fois autonome et extensible. En plus d'utiliser ses LED intégrées, boutons et capteurs, nous pouvons élargir sa gamme de fonctions en l'insérant dans un connecteur.

On la programme avec MuPython (<https://codewith.mu/>) ou en ligne sur Vittascience (<https://fr.vittascience.com/microbit/>).



2 Programmation

- Ouvrir l'éditeur, et copier-coller le programme ci-dessous :

```
from microbit import *
while True:
    if button_a.is_pressed() :
        display.show(Image.HAPPY)
```

L'instruction `display.show('1')` affiche le caractère '1' et `display.show(Image.SAD)` affiche le smiley triste. `button_a.is_pressed()` renvoie `TRUE` si a est pressé, `FALSE` sinon.

- Connectez la carte à un port USB de votre PC.

- Transférer le code sur la carte :

Sur MuCode → Flasher la carte.

Sur Vittasciences → Upload le programme exécutable (microbit.hex), à transférer dans la carte micro:bit

Une DEL doit clignoter au dos de celle-ci lors du transfert. Lorsque celui-ci est terminé, le programme est utilisable.

La documentation est disponible ici : <https://microbit-micropython.readthedocs.io/fr/latest/>

3 Les fonctions logiques de base

Avec les deux entrées TOR (Tout Ou Rien), le bouton A, le bouton B, qui ne peuvent prendre chacun que deux valeurs booléennes False ou True nous pouvons programmer et tester des fonctions de logique booléenne.

1^{ère} opération logique

- Créer un programme qui affiche « 0 » si a est pressé, et « 1 » sinon.
- Complétez alors la **table de vérité** de la fonction logique ainsi créée :

A	S
0	
1	

On admettra provisoirement que l'expression booléenne est de la forme : $S = \bar{A}$

2nde opération logique

- Créer un programme qui affiche « 1 » si a et b sont pressés, et « 0 » sinon.
- Complétez alors la **table de vérité** de la fonction logique ainsi créée :

A	B	S (S = Sortie)
0	0	
0	1	
1	0	
1	1	

On admettra provisoirement que l'expression booléenne est de la forme : $S = A.B$

2^{ème} opération logique

- Créer un programme qui affiche « 1 » si a ou b sont pressés, et « 0 » si aucun des 2 ne l'est.
- Complétez alors la **table de vérité** de la fonction logique ainsi créée :

A	B	S (S = Sortie)
0	0	
0	1	
1	0	
1	1	

On admettra provisoirement que l'expression booléenne est de la forme : $S = A+B$

- Conclure sur les nom que l'on peut donner à ces 3 fonctions logiques.

4 Les fonctions plus avancées

- Programmer chacune à leur tour les fonctions logiques suivantes et complétez leur table de vérité respective :

non (A et B)

A	B	S
0	0	
0	1	
1	0	
1	1	

non A ou non B

A	B	S
0	0	
0	1	
1	0	
1	1	

non (A ou B)

A	B	S
0	0	
0	1	
1	0	
1	1	

non A et non B

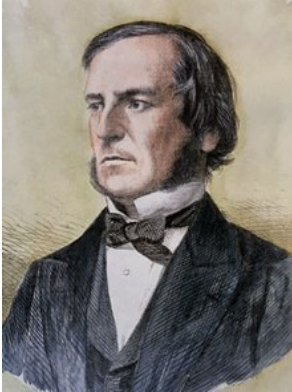
A	B	S
0	0	
0	1	
1	0	
1	1	

- Complétez la table de vérité et programmer le fonctionnement d'un système dont l'équation serait :

$$S = \overline{A \cdot B} + A \cdot \overline{B}$$

A	B	S
0	0	
0	1	
1	0	
1	1	

ANNEXE Doc historique : L'algèbre de Boole



George Boole est un logicien, mathématicien et philosophe britannique. De 1844 à 1854, il crée une algèbre binaire, dite booléenne, n'acceptant que deux valeurs numériques : 0 et 1 que l'on appelle algèbre de Boole en son honneur !



Augustus De Morgan est un mathématicien et logicien britannique. Il est le fondateur avec Boole de la logique moderne (1842-1864). Il a notamment formulé les lois qui porte son nom.



Claude Elwood Shannon est un ingénieur en génie électrique et mathématicien américain. Il est le père fondateur, de la théorie de l'information.
1938 : publication de *A Symbolic Analysis of Relay and Switching Circuits*. Shannon applique pour la première fois l'algèbre de Boole aux circuits de commutation automatique, qui servira aussi bien pour les circuits à relais que pour les circuits intégrés.