

Introduction

Cette brochure propose des activités permettant de prendre en main les appareils micro:bit. Cette première partie ne demande pas la connexion de périphérique externe pour la réalisation des activités.

Les activités sont organisées progressivement par ordre de difficulté en ajoutant petit à petit des compétences nécessaires à la réussite des défis.

Des ressources vidéos sont également disponible en ligne pour chaque activité, le contenu multimédia est disponible dans la section micro:bit de l'adresse suivante:

<https://edu.ge.ch/site/robotic>

Codes QR



Site RoboTIC
<https://edu.ge.ch/site/robotic>



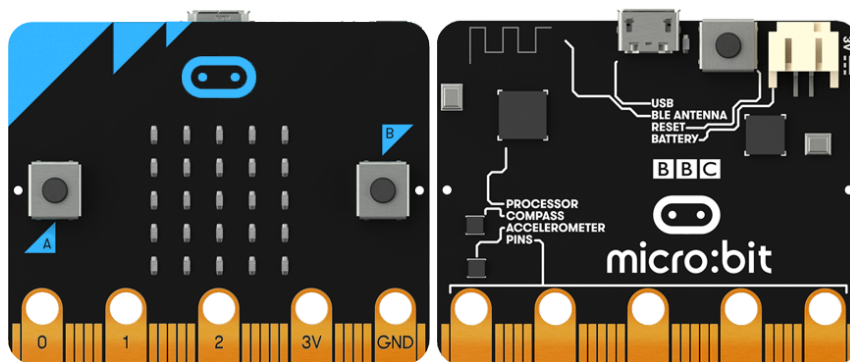
Section micro:bit du site RoboTIC
<https://edu.ge.ch/site/robotic/category/ordinateur-monocarte/micro-bit/>

Certaines activités proposent des prolongements, offrant des options de développement plus complexe de la tâche demandée. Ces prolongement ont pour but de complexifier les programmes et amener les élèves à utiliser des formes de programmation plus évoluées en exploitant notamment des structures de contrôle et des variables.

Micro:bit - Les bases

Le micro:bit est un ordinateur à carte unique comprenant différents éléments tels que:

- Bluetooth
- Compas
- Accéléromètre
- 2 boutons programmables
- Connecteur micro USB
- Connecteur de batterie 3V
- 25 LEDs programmables
- 25 connecteurs
 - 5 grandes broches
 - 20 petites broches



Le micro:bit est capable d'exécuter deux langages de programmation: le Python et le JavaScript. Pour créer des programmes exécutables par le micro:bit, deux consoles de programmation en ligne sont accessibles à ces adresses:

- Éditeur Python: python.microbit.org
- Éditeur JavaScript: makecode.microbit.org

L'éditeur utilisé dans les défis à venir est l'éditeur JavaScript car il permet à la fois une programmation visuelle par blocs ainsi qu'une programmation textuelle en langage "brut".

Utilisation

Pour faire exécuter un programme par le micro:bit, il est d'abord nécessaire de compiler et transférer le programme depuis la console de programmation. Une fois le fichier récupéré, il suffit de le transférer dans la mémoire du micro:bit via une connexion USB.

Interface de programmation

L'interface de programmation visuelle est composée de trois zones principales permettant de construire un programme et de vérifier son exécution. Cette interface interactive inclut également un contrôle d'erreur en temps réel qui avertit l'utilisateur des éventuelles incohérences contenues dans le programme.

Principales zones de l'interface

L'interface propose: un Simulateur, une Boîte à outils et un Espace de travail
Les interactions entre la Boîte à outils et l'espace de travail se font par glisser-déposer.



La zone inférieure de l'interface de programmation permet de nommer le programme ainsi que de le télécharger sur l'ordinateur. Le fichier téléchargé est en format `.hex` et doit ensuite être copié sur l'appareil micro:bit "physique" ([exemple en ligne](#), en anglais avec vidéo).

En cas de difficulté, une page de support est disponible en ligne (en anglais):

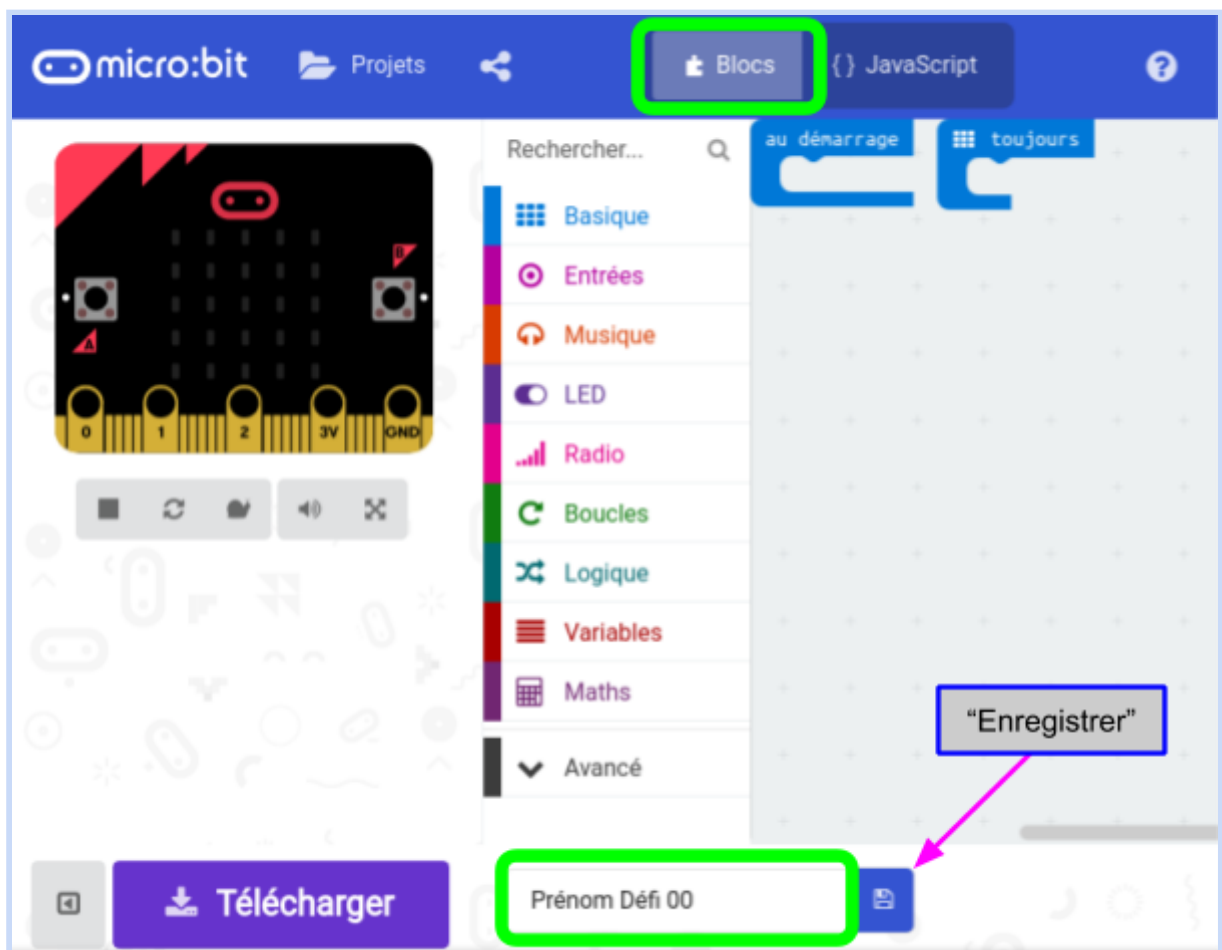
<https://support.microbit.org/support/solutions>

Micro:bit - Tutoriel 00

Le programme suivant permettra d'afficher en alternance deux icônes de votre choix pour une durée indéterminée (l'exemple ci-dessous affichera l'icône "Coeur" et l'icône "Heureux" en alternance). Il affichera également la lettre "A" lorsque le bouton A est pressé et la lettre "B" lorsque le bouton B est pressé.

Etape 1

Depuis le mode "Blocs" de la console de programmation makecode.microbit.org, créer un nouveau programme nommé "**Prénom Défi 00**" puis cliquer sur "Enregistrer".



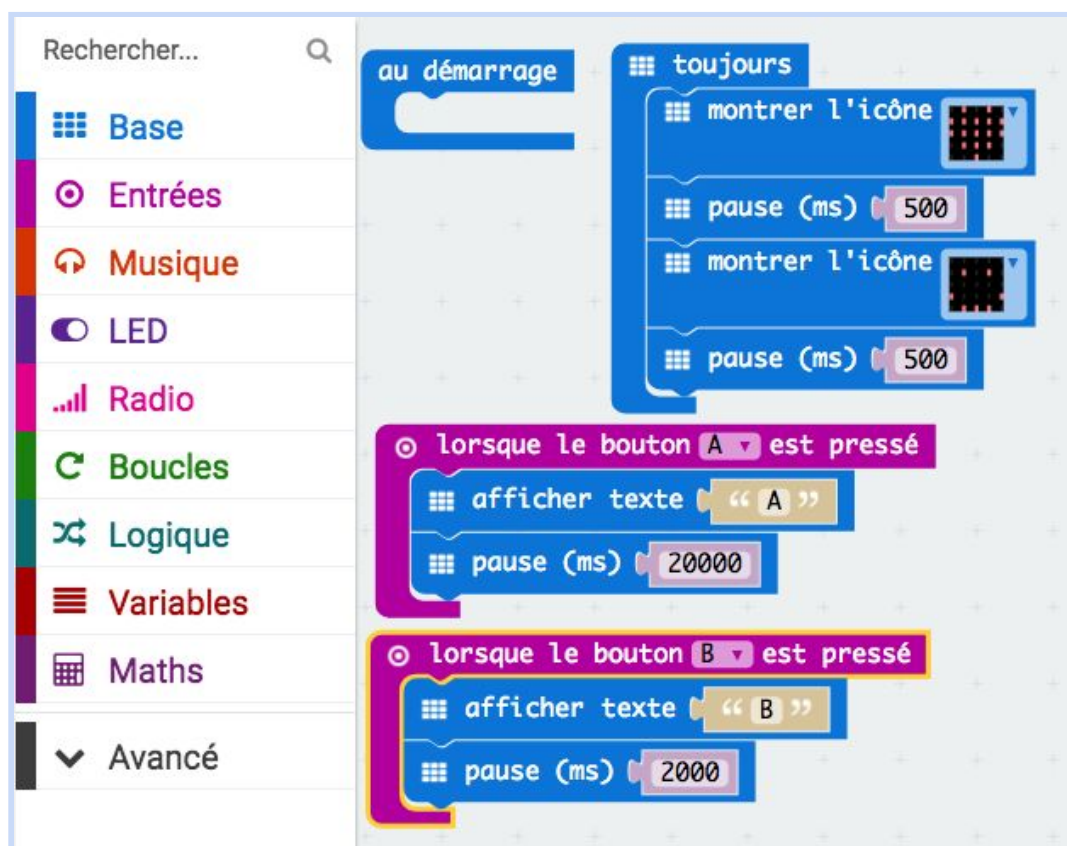
Le nommage des programmes est une bonne pratique à adopter afin de pouvoir retrouver facilement les travaux depuis le répertoire "Projets" de la console de programmation ainsi que dans les fichiers téléchargés.

Etape 2

Glisser les blocs de programmation correspondant pour obtenir le résultat illustré ci-dessous.

- Depuis la catégorie “Base”, glisser les blocs “montrer l’icône” adéquats dans le bloc “toujours” et leur attribuer les icônes “Coeur” et “Heureux”
- Depuis la catégorie “Entrées”, glisser deux blocs “lorsque le bouton A est pressé” sur l’espace de travail
 - *Modifier la variable “A” en variable “B” pour l’un des deux blocs, comme sur l’exemple ci-dessous*
- Depuis la catégorie “Basique”, glisser les blocs “afficher texte” dans les blocs “lorsque le bouton ... est pressé”
 - Modifier les variables pour correspondre à l’exemple
- Depuis la catégorie “Basique”, glisser les blocs “pause (ms)” à la suite des blocs “afficher texte” et “montrer icône”
 - *Modifier les variables des blocs à “2000”*
- Enregistrer votre projet

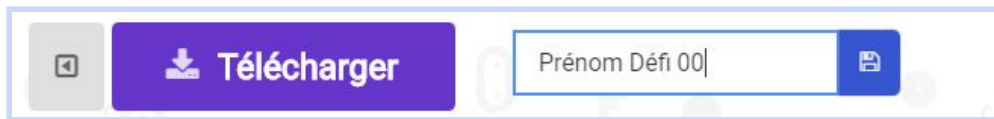
Exemple du programme à réaliser



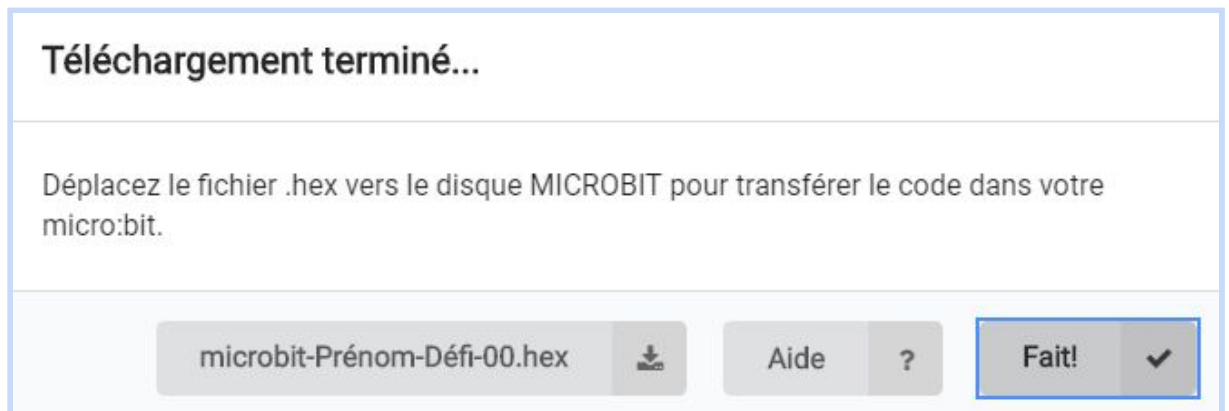
Etape 3

Le programme étant terminé, il faut le compiler et le télécharger afin de déposer le fichier programme `.hex` sur le micro:bit.

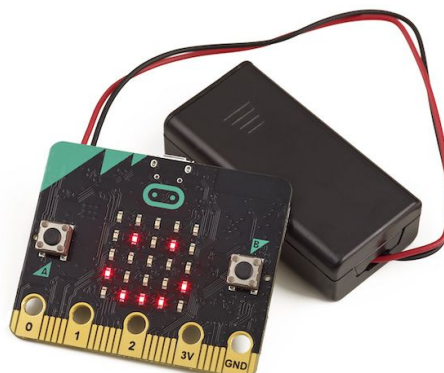
1. Cliquer sur le bouton "Télécharger"



2. Confirmer le téléchargement du fichier `.hex`



3. Depuis le navigateur de fichiers de l'ordinateur, copier-coller le fichier sur le micro:bit ([exemple en ligne](#), en anglais avec vidéo).
4. Après le chargement du fichier sur l'appareil, ce dernier exécute le programme créé et le conserve dans sa mémoire interne jusqu'à ce qu'un autre fichier programme `.hex` ne soit copié (un seul programme à la fois peut être stocké sur l'appareil).
5. Afin de vérifier le bon fonctionnement du programme, déconnecter le micro:bit de l'ordinateur et le mettre sous tension avec la batterie 3V fournie ([exemple en ligne](#), en anglais).

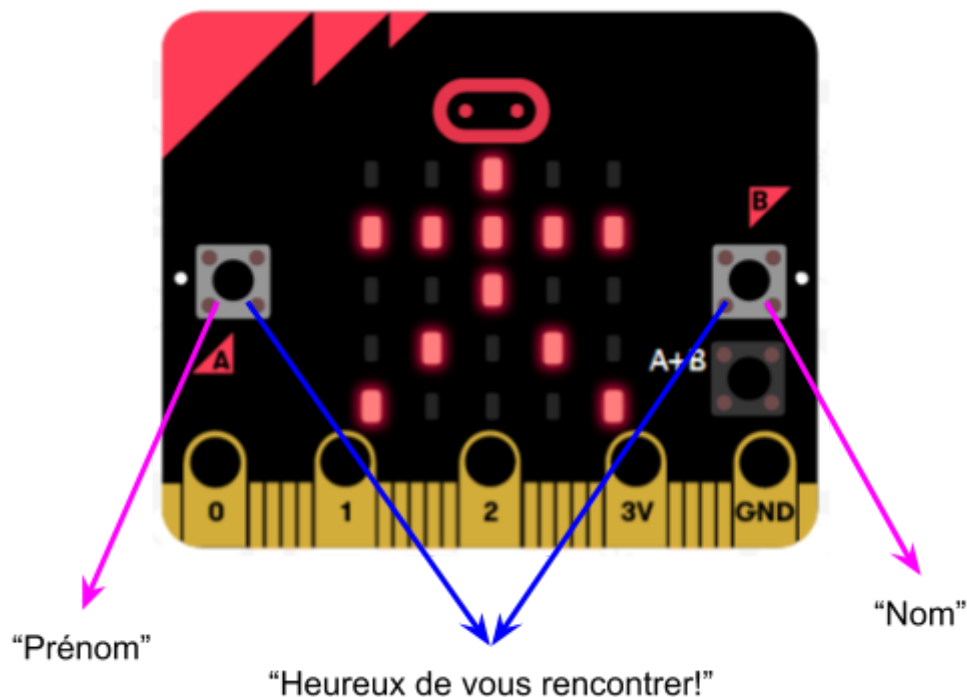


Micro:bit - Défi 01

Créer un programme nommé “**Prénom Défi 01**” avec l’interface de programmation “Blocs” de makecode.microbit.org.

Le programme devra permettre au micro:bit d’afficher votre prénom lorsque la touche “A” est pressée et afficher votre nom de famille lorsque la touche “B” est pressée.

De plus, il devra afficher l’icône “Bonhomme” au démarrage du programme et afficher le texte “Heureux de vous rencontrer!” lorsque les boutons A et B sont pressés simultanément.



Exemple vidéo

<https://edu.ge.ch/site/robotic/wp-content/uploads/sites/74/2019/11/Défi-01-microbit.mp4>

Une fois terminé, bien penser à **conserver une copie du programme** dans un emplacement sûr et facile à retrouver (un dossier sur l’ordinateur ou une clé USB par exemple).

Micro:bit - Défi 02

Créer un programme nommé “**VotrePrénom Défi 02**” avec l’interface de programmation “Blocs” de makecode.microbit.org.

Le programme devra permettre au micro:bit de démarrer et d’afficher un compte à rebours de 5 à 0 secondes lorsque la touche “A” est pressée. L’affichage d’au moins 5 LEDs de votre choix au démarrage du programme doit également faire partie de la séquence exécutée sur le micro:bit. Une fois le compte à rebours terminé, une icône de votre choix devra clignoter 3 fois sur l’écran de l’appareil avant d’effacer le contenu de l’écran.

Exemple vidéo

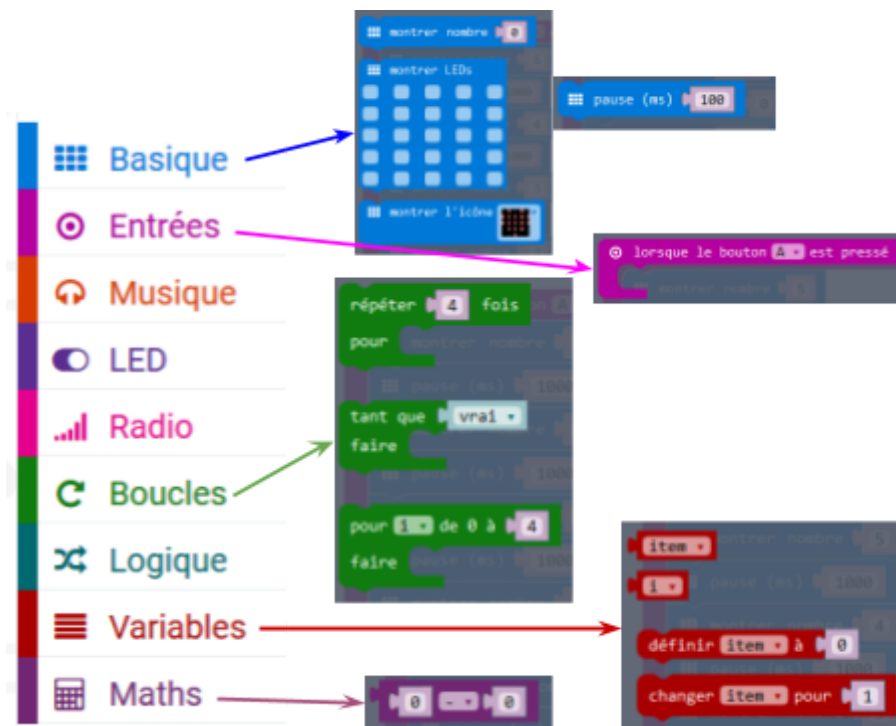
ATTENTION! votre compte à rebours doit décompter des secondes, il faudra donc ajouter les bons blocs de programmation pour correspondre à cette contrainte!

Afin d’obtenir un **bonus** pour votre travail, optimisez votre programme au moyen de boucles de programmation, de variables et d’opération mathématiques!

Informations utiles

Pour ce défi, les outils de la section “**Basique**” et “**Entrées**” sont suffisant. Cependant, l’obtention du bonus nécessite au moins l’utilisation des outils “**Boucles**” et “**Variables**”.

Une **boucle** permet de répéter des opérations un certain nombre de fois selon certaines conditions
Une **variable** est un nombre qui peut changer de valeur durant l’exécution d’un programme



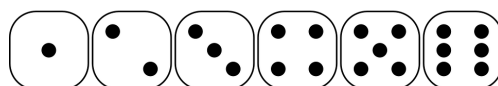
Une fois terminé, bien penser à **conserver une copie du programme** dans un emplacement sûr et facile à retrouver (un dossier sur l’ordinateur ou une clé USB par exemple).

Micro:bit - Défi 03

Créer un programme nommé “**VotrePrénom Défi 03**” avec l’interface de programmation “Blocs” de makecode.microbit.org.

Le programme devra permettre au micro:bit de simuler l’utilisation d’un dé à 6 faces. Il s’agit donc de créer un programme qui affichera de façon aléatoire une des 6 faces d’un dé après avoir secoué l’appareil.

Au démarrage du programme, les 6 faces doivent s’afficher successivement et montrer une icône de votre choix pour indiquer que l’appareil est prêt à lancer les dés! Cette séquence de démarrage doit également pouvoir être jouée en appuyant sur le bouton “A”.



Bonus

Lors du “jet” de dés, afficher une séquence montrant différentes faces du dés avant d’afficher le résultat!

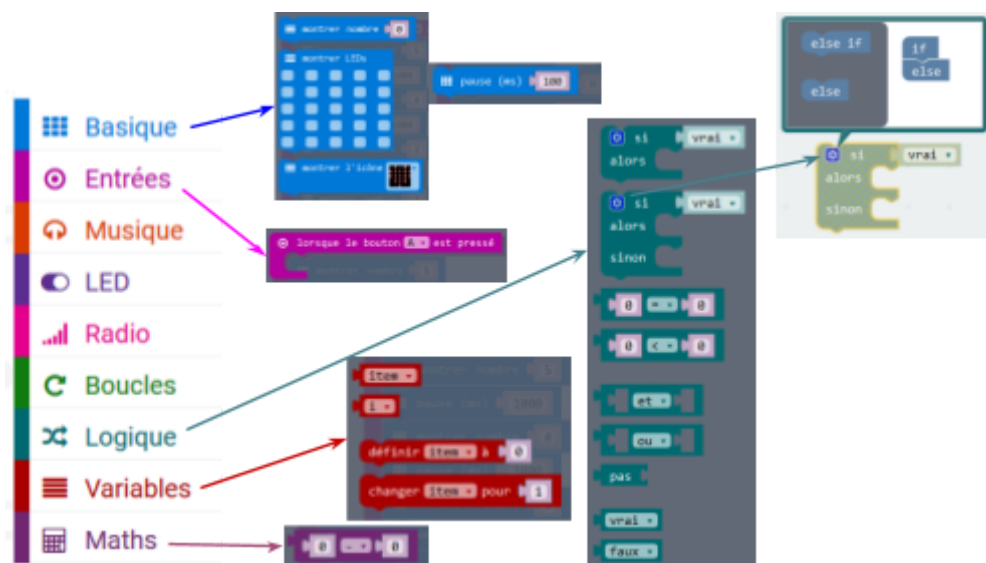
[Exemple Vidéo Défi 03 avec Bonus](#)

Informations utiles

Pour ce défi, les outils de la section “**Basique**”, “**Entrées**”, “**Logique**”, “**Maths**” et “**Variables**” sont utiles.

Une opération logique permet de réaliser une action selon une condition de type “si ..., ou si ..., alors”. Dans ce défi, cette opération permet d’afficher les LEDs correspondant à un nombre (une variable).

Une **variable** est un nombre qui peut changer de valeur durant l’exécution d’un programme



Une fois terminé, bien penser à **conserver une copie du programme** dans un emplacement sûr et facile à retrouver (un dossier sur l’ordinateur ou une clé USB par exemple).

Micro:bit - Défi 04

Créer un programme nommé “**VotrePrénom Défi 04**” avec l’interface de programmation “Blocs” de makecode.microbit.org.

Le programme à créer doit permettre à l’utilisateur de l’appareil d’envoyer et de recevoir des signaux dans un groupe radio. Pour cela, il faut s’accorder sur le numéro du groupe radio à utiliser avec les futurs membres du groupe avant ou pendant la création du programme afin que tous les membres puissent échanger des informations.

L’appareil devra également réagir aux signaux reçus par les autres membres du groupe en affichant une icône selon la donnée reçue. Afin de simplifier et de standardiser les échanges d’information entre les appareils, certaines conditions sont à respecter. Chaque touche (ou combinaison de touches) doit transmettre une valeur différente de cette façon:

Touche “A” -> envoie 0; Touche “B” -> envoie 1; Touche “A+B” -> envoie 2

Afin de rendre le groupe interactif, l’appareil devra répondre aux sollicitations des autres membres en affichant une icône correspondant à chaque valeur reçue (0, 1 ou 2). Comme chaque appareil est unique, vous êtes libre de choisir les icônes que vous souhaitez afficher sur votre appareil lors de la réception des données!

[Exemple vidéo Défi 04](#)

Informations utiles

Pour ce défi, explorez les différentes sections de la boîte à outil afin de combiner l’utilisation des fonctions que vous connaissez telles que “**Basique**”, “**Entrées**” et “**Logique**”. Ce défi nécessite en plus l’utilisation des outils “**Radio**” afin de définir un groupe radio et de gérer l’événement “*donnée reçue par radio*”. Une distinction est faite entre deux types de données: numérique et littérale. Afin que votre programme fonctionne, soyez bien attentif au **type de donnée** que vous traitez!



Il est indispensable d’effectuer des **tests** sur les données traitées, utilisez donc judicieusement les structures de contrôle adéquate!

Une fois terminé, bien penser à **conserver une copie du programme** dans un emplacement sûr et facile à retrouver (un dossier sur l’ordinateur ou une clé USB par exemple).

Proposition de prolongement

Bataille de dés

Imaginer une combinaison de ce défi et du défi 03 (tirage de dés) pour créer une “bataille de dés” entre les membres du groupe, chaque appareil doit générer un nombre correspondant aux dés puis comparer son résultat avec les autres membres du groupe afin de déterminer s’il a gagné ou non. Il est utile d’ajouter une pause d’environ 1 seconde avant d’envoyer la donnée par radio pour laisser le temps aux autres joueurs d’effectuer leur tirage!

Ce prolongement nécessite d’ajouter des tests entre les variables envoyées et reçues afin de déterminer le gagnant!

Afin de limiter la difficulté, ce prolongement peut se faire avec un groupe d’uniquement deux joueurs.

Micro:bit - Défi 05

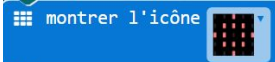
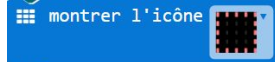
Créer un programme nommé “**VotrePrénom Défi 05**” avec l’interface de programmation “Blocs” de makecode.microbit.org.

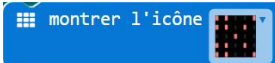
Le programme à créer doit permettre à l’utilisateur de jouer à feuille-cailloux-ciseaux avec un adversaire. Pour cela, il faut s’accorder sur le numéro du groupe radio à utiliser avec votre partenaire de jeu avant la création du programme afin que les deux joueurs puissent échanger des informations.

Au démarrage, l’appareil doit indiquer qu’il est prêt en montrant l’icône  (oui). Ensuite, un nombre de 0 à 2 doit être généré aléatoirement lorsque l’on secoue l’appareil afin d’afficher le symbole joué. Cette information doit ensuite être envoyée par radio au joueur adverse afin de déterminer le gagnant. Le gagnant doit afficher l’icône “heureux” alors que le perdant devra afficher l’icône “triste” et ce pendant 1 secondes chacun. Pour faire un nouveau duel, les deux joueurs doivent appuyer sur le bouton “A” de leur appareil.

Exemple Vidéo Défi 05

Afin d’avoir un jeu cohérent, les valeurs numériques doivent correspondre aux mêmes icônes:

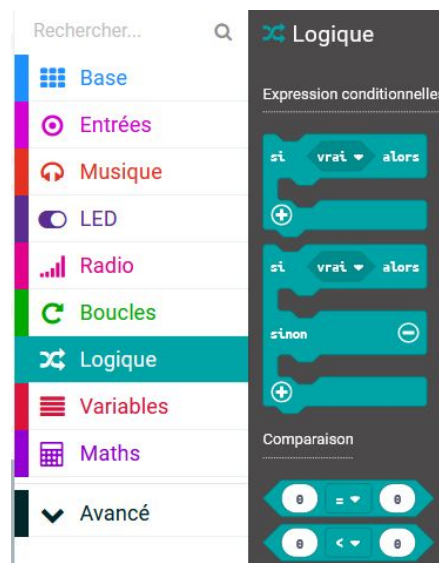
Feuille = 0  **Cailloux = 1** 

Ciseaux = 2 

Informations utiles

Pour ce défi, explorer les différentes sections de la boîte à outil afin de combiner l’utilisation des fonctions que vous connaissez telles que “**Basique**”, “**Entrées**”, “**Logique**”, “**Boucles**” ou “**Variables**”, vous aurez également besoin des outils “**Radio**”.

Il est nécessaire d’ajouter des tests entre les variables envoyées et reçues afin de déterminer le gagnant!



Une fois terminé, bien penser à **conserver une copie du programme** dans un emplacement sûr et facile à retrouver (un dossier sur l’ordinateur ou une clé USB par exemple).

Propositions de prolongement

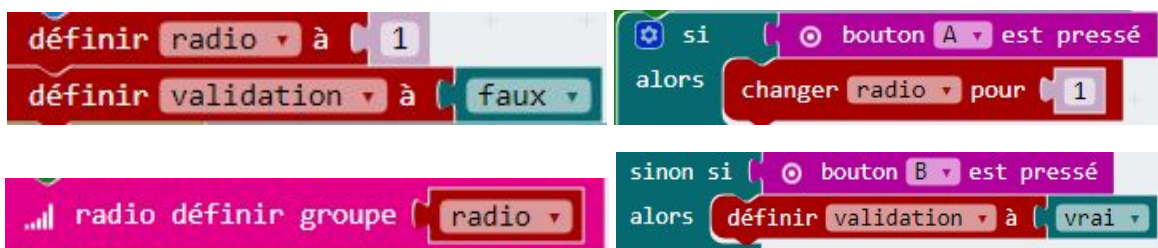
Compter les points

Afin de perfectionner le jeu, créer un système d’affichage du score permettant de voir combien de duels ont été gagnés par l’appareil! Vous devrez également mettre en place un moyen de remettre le score à zéro pour commencer une nouvelle partie.

Rejoindre un groupe radio (ou salle de jeu)

Pour cela, il doit être possible de choisir le groupe radio au démarrage de l’appareil afin de rejoindre un groupe de jeu. Le démarrage du programme devra inviter l’utilisateur à choisir le groupe radio en appuyant sur la touche “A”: chaque pression incrémente le numéro du groupe de 1. Ensuite, l’utilisateur doit pouvoir valider son choix en pressant la touche “B” et la première lettre de votre prénom doit apparaître.

Il est nécessaire de créer différentes variables pour ce défi, il est recommandé de créer une **variable numérique** pour le choix du groupe radio ainsi qu’une **variable booléenne** pour définir l’état de validation du groupe radio (vrai/faux).



[Exemple vidéo Défi 05 avec sélection du groupe radio](#)

Micro:bit - Défi 06

Créer un programme nommé **“VotrePrénom Défi 05”** avec l’interface de programmation “Blocs” de makecode.microbit.org.

Le programme à créer doit permettre à l’utilisateur de l’appareil de s’orienter grâce à la boussole de l’appareil et également de pouvoir consulter la température du microprocesseur du micro:bit.

Pour cela, lors du démarrage de l’appareil les indications d’utilisations doivent apparaître:

- Appuyer sur le bouton **“A”** pour afficher la température du microprocesseur
- Appuyer sur le bouton **“B”** pour afficher l’orientation (Nord, Est, Sud, Ouest)

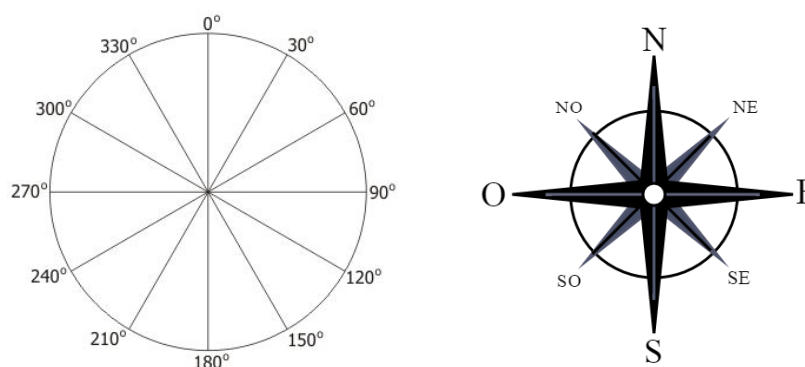
La fin de la séquence de démarrage doit afficher un symbole de flèche double (gauche-droite) que vous devrez créer et faire clignoter 3 fois. Il faudra également pouvoir rejouer la séquence de démarrage en appuyant simultanément sur les bouton **“A+B”**.

Afin d’améliorer l’expérience utilisateur de la boussole, il faudra que celle-ci affiche **de manière continue l’orientation tant que le bouton “B” est pressé**.

[Exemple vidéo Défi 05](#)

Informations utiles

La boussole du micro:bit renvoie ses informations sous forme numérique de **0 à 359** degrés (un tour complet). Il faut donc faire correspondre ces données numérique à une information textuelle pour orienter l’utilisateur (la valeur 0° correspond au Nord). Lors de la première utilisation de la boussole, l’appareil demandera de dessiner un cercle afin de calibrer la boussole ([exemple vidéo calibration boussole](#)).



Pour ce défi, explorer les différentes sections de la boîte à outil afin de combiner l’utilisation des fonctions que vous connaissez telles que **“Basique”**, **“Entrées”**, **“Logique”**, **“Boucles”** ou **“Variables”**.

L’expression “de manière continue” correspond à une boucle de programmation “pour toujours”.

Bonus

Afin de perfectionner votre appareil, affinez les informations d’orientation en ajoutant les données: Nord Est (NE), Sud Est (SE), Sud Ouest (SO) et Nord Ouest (NO).

Ajouter également une icône selon la température affichée:

- < 30° afficher: :-)
- ≥ 30° afficher: :-(