



## IEEE Ottawa Robotics Competition Compétition de robotique d'Ottawa d'IEEE

# Défi Mix:Tape

Dernière Révision: 5 février 2023

## Table des matières

Défi Mix:Tape	2
Simulateur Micro:Bit	2
Langues de programmation	3
Règlements du défi	4
Évaluation	5
Soumission	5

### Avis

Vous avez la responsabilité de lire et de comprendre ce document de façon régulière car il se peut que nous la modifions périodiquement.

Si vous avez des questions, veuillez contacter  
notre équipe Micro:Bit à  
[orcinfo@ieeeottawa.ca](mailto:orcinfo@ieeeottawa.ca)

## **Défi Mix:Tape**

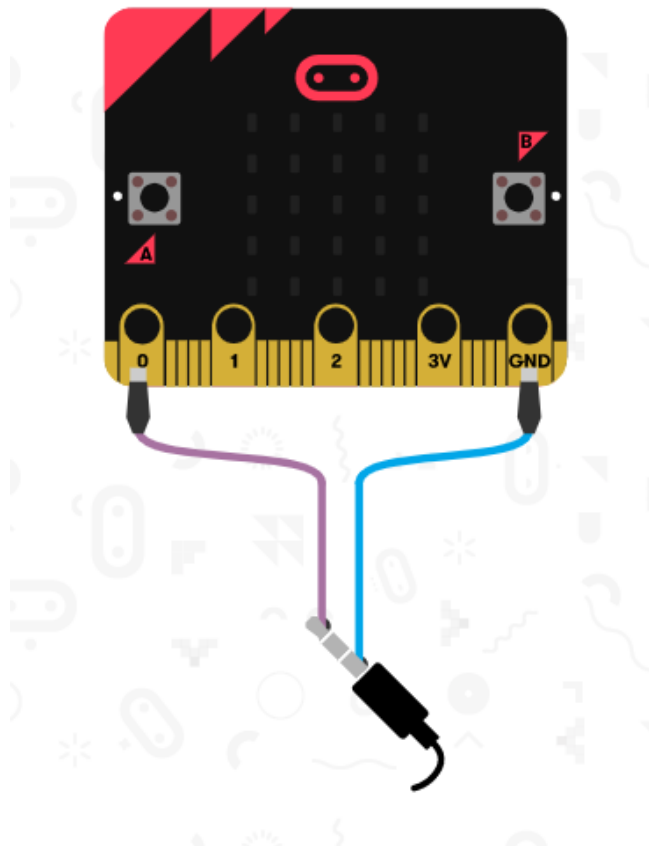
Dans les années 1970, les gens aimaient créer des collections de musique sur des cassettes et les partager entre eux pour que leurs amis les apprécient. Votre tâche consiste à utiliser un simulateur Micro:bit pour fabriquer une mixtape robotique : un lecteur de musique capable de lire une collection de chansons accompagnée de visuels LED. Vous pouvez même la partager avec vos amis !

## **Simulateur Micro:Bit**

Pour ce défi, vous utiliserez un simulateur Micro:Bit gratuit et disponible en ligne. Vous pouvez y accéder à travers ce lien

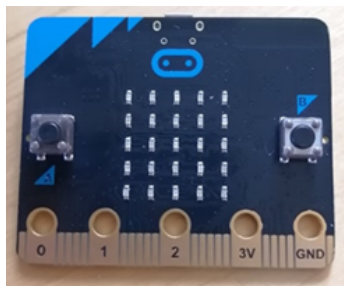
<https://makecode.microbit.org/> ou télécharger l'application via le Microsoft Store. Vous pouvez aussi télécharger l'application iOS ou Android mais nous vous recommandons d'utiliser cette option uniquement si vous avez une grande tablette:

<https://www.microsoft.com/en-ca/p/makecode-for-micro-bit/9pjc7sv48lcx?activetab=pivot:overviewtab>



**Image 1: Simulateur Micro:Bit**

Cependant, si vous avez déjà acheté le kit, vous pouvez utiliser votre Micro:Bit physique pour tester votre code. Ce n'est pas nécessairement recommandé, car vous pouvez trouver cette option plus lente que d'utiliser simplement le simulateur dans le logiciel. Si vous décidez d'utiliser votre Micro:Bit, le seul composant pertinent sera la puce Micro:Bit. Que vous décidiez d'utiliser le simulateur ou le kit physique, vous soumettrez seulement votre code pour l'évaluation et la notation et nous testerons votre code sur le simulateur. Vous devez cependant garder à l'esprit que dans le contexte du simulateur, il est impossible d'appuyer sur deux boutons à la fois. Vous pouvez également utiliser les broches comme boutons lors de l'utilisation du simulateur.



**Image 2: Micro:Bit**

## Langues de programmation

L'éditeur MakeCode permet de programmer en blocs de code, Javascript ou Python. Vous pouvez utiliser n'importe lequel de ces trois, mais puisque notre jugement sera effectué dans l'environnement des blocs de codage, il sera de votre responsabilité de vous assurer que votre code fonctionne à la fois dans cet environnement et qu'il est bien organisé.

Il y a quelques incohérences lors du passage d'un environnement de codage à un autre. Lorsque vous passez de Javascript ou de Python à l'environnement des blocs de codage, il tentera de transformer vos mélodies en représentation de blocs de la mélodie, qui est limitée à une seule octave, n'a pas de dièses et de bémols et ne dure que 8 notes. Cependant, il ne tentera de le transformer en cette représentation que si vous placez directement la chaîne musicale dans la fonction "jouer la mélodie". Si vous l'enregistrez en tant que variable de texte séparément, tout fonctionnera. L'utilisation de variables appropriées pour les mélodies est le moyen le plus simple de s'assurer que les mélodies ne seront pas transformées lors du passage de Javascript ou Python aux blocs de code.

Si vous ne l'enregistrez pas séparément en tant que variable de texte, il tentera de la transformer. Si vous n'indiquez pas de bémols et de dièses et que votre mélodie n'utilise pas de notes qui sont en dehors de l'octave des blocs de code, elle réussira à la transformer et coupera également votre mélodie à 8 blocs de long.

L'environnement des blocs de code peut également utiliser ce comportement. Vous pouvez jouer des mélodies plus complexes en sauvegardant la mélodie dans une variable de texte et l'utiliser dans la fonction "jouer la mélodie".

La documentation est disponible ici:

<https://makecode.microbit.org/reference>

## **Règlements du défi**

1. Votre lecteur de musique doit inclure au moins 3 mélodies avec une icône différente affichée sur l'écran LED pour chaque mélodie.
2. Les mélodies doivent durer au maximum 2 minutes.
3. Votre lecteur de musique doit jouer chaque mélodie en ordre et retourner à la première mélodie après la dernière.
4. Votre lecteur de musique doit avoir au moins deux fonctionnalités à l'aide des boutons (par exemple, augmenter/réduire le volume, couper/rétablir le son, retourner à la première mélodie, mélodie suivante/précédente).
5. Si votre code produit un message d'erreur au démarrage du programme, vous obtiendrez automatiquement un score de 0. S'il produit un message d'erreur seulement pour certaines fonctionnalités, des points seront déduits.

Oups, nous n'avons pas réussi à exécuter ce projet. Veuillez rechercher les erreurs dans votre code.

6. Vous devez soumettre un manuel d'instructions qui explique le fonctionnement de votre lecteur de musique, avec un diagramme de quels boutons sur le Micro:Bit sont liées à quelles fonctionnalités, ainsi

que toute information supplémentaire que vous jugez nécessaire pour comprendre comment utiliser votre lecteur de musique.

7. Le manuel d'instruction ne devrait pas être plus long qu'une page. Il peut être soit créé de façon digitale ou écrit à la main et puis scanné, à condition que le texte soit lisible. S'il ne peut pas être lu, vous recevrez un score de 0 pour la section.
8. Toutes les fonctions qui fonctionnent dans le simulateur sont autorisées, puisque nous utilisons le simulateur. La fonctionnalité radio permettant d'avoir plus d'un Micro:Bit simulé est également explicitement autorisée.

## Évaluation

1. Votre code sera évalué basé sur les concepts de code de bonne qualité. Ceci inclut l'organisation visuelle des blocs de code, aucune réutilisation de lignes de code, une bonne utilisation des fonctions/boucles de programmation, avoir des noms descriptifs pour vos fonctions et vos variables, et une bonne utilisation de commentaires de code. Le jugement sera effectué dans le format des blocs de codage.
2. Votre lecteur de musique sera jugé sur sa créativité et son originalité. Il devrait inclure des mélodies et des icônes LED programmées par vous. Vous pouvez utiliser de vraies chansons et vous n'aurez pas besoin de tout composer à partir de zéro, mais vous serez pénalisé si vous utilisez les mélodies prédéfinies.
3. Tous les scores tiendront compte de votre niveau scolaire.
4. Le gagnant du défi sera déterminé par le score combiné de votre manuel d'instruction, de votre rapport d'une page, des fonctionnalités de votre programme, de la qualité du code et de votre créativité et originalité. L'équipe avec le score combiné le plus élevé sera le gagnant du défi.
5. Vous pouvez gagner des points bonus si vous incluez plus de 2 fonctionnalités et/ou si les mélodies sont basées sur un thème décrit dans le rapport.
6. Les décisions des juges sont définitives.

## **Soumission**

Pour soumettre votre travail, vous devez remplir un formulaire fourni dans le lien ci-dessous. Ce formulaire nécessite le nom de votre équipe, votre code sous forme de fichier .hex, votre manuel d'instructions qui peut être écrit à la main ou tapé et le rapport d'une page qui doit être tapé. Le manuel d'instructions et le rapport doivent être soumis sous forme de fichier .pdf. Le lien pour accéder au formulaire de soumission se trouve ici:

<http://www.orc.ieeeottawa.ca/soumission/>