

Situation déclenchante de l'activité

Une zone hautement radioactive est devenue inaccessible à l'homme après une catastrophe. Seul un robot autonome peut y accéder.

Une agence de sûreté nucléaire vous missionne : concevoir, adapter et programmer un robot pour aller chercher un fût radioactif et le déposer dans une zone de décontamination sécurisée.

Ce scénario correspond à l'épreuve 1 du challenge robotique organisé avec d'autres collèges de la ZAP.



Fukushima : des robots au cœur de l'enfer :

<https://youtube.com/clip/UgkxOv3Oot7ojKHdV4e4k1U47QXXX3HMc0oU?si=vDYeKUwYuBjrgewB>

Votre robot mBot doit :

- ☒ suivre une ligne noire,
- ☒ éviter un obstacle (cube),
- ☒ repérer et transporter un fût (cylindre),
- ☒ atteindre une boîte de décontamination (zone cible fermée).

Mes constats, mes observations

Mon problème à résoudre

Mes idées pour le résoudre

Les idées retenues pour résoudre le problème

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

Situation déclenchante de l'activité

Une zone hautement radioactive est devenue inaccessible à l'homme après une catastrophe. Seul un robot autonome peut y accéder.

Une agence de sûreté nucléaire vous missionne : concevoir, adapter et programmer un robot pour aller chercher un fût radioactif et le déposer dans une zone de décontamination sécurisée.

Ce scénario correspond à l'épreuve 1 du challenge robotique organisé avec d'autres collèges de la ZAP.



Fukushima : des robots au cœur de l'enfer :

<https://youtube.com/clip/UgkxOv3Oot7ojKHDV4e4k1U47QXXX3HMc0oU?si=vDYeKUwYuBjrgewB>

Votre robot mBot doit :

- ✓ suivre une ligne noire,
- ✓ éviter un obstacle (cube),
- ✓ repérer et transporter un fût (cylindre),
- ✓ atteindre une boîte de décontamination (zone cible fermée).

Mes constats, mes observations

- Utiliser un robot autonome capable de détecter et suivre un parcours
- Concevoir une pièce pour pousser le fût sans le toucher directement
- Programmer le robot pour contourner les obstacles

Mon problème à résoudre

« Comment programmer un robot pour le challenge robotique et ainsi accéder à un endroit trop dangereux pour l'homme ? »

Mes idées pour le résoudre

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

On utilise un robot mBot programmé pour suivre un parcours, contourner des obstacles et déplacer un fût vers une zone sécurisée.

Les idées retenues pour résoudre le problème

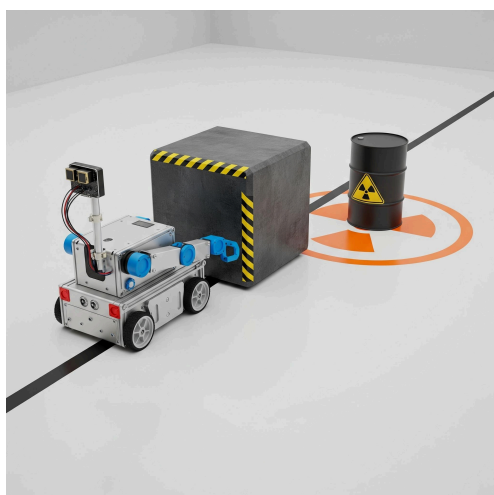
On utilise un robot pour rentrer dans les décombres de la centrale.

Compétences et connaissances travaillées dans l'activité

Thème abordé : CCRI Attendu de fin de cycle : CCRI3-Concevoir, écrire, tester et mettre au point un programme Thématique : T22 - La programmation des OST	
Compétence CCRI31 - Modifier un algorithme permettant de répondre au besoin ou au problème posé.	Connaissances SFC3c - instruction conditionnelle. SFC3d-Algorithmique et programmation : instructions itératives OST2b - les modes de représentation : croquis, schéma, graphique, algorithme, modélisation.
Critères d'apprentissages	N1 – Je sais reconnaître et expliquer ce qu'est un algorithme.
	N2 – et je sais justifier l'utilisation d'un algorithme
	N3 – et je sais modifier un algorithme permettant de répondre au besoin ou au problème posé.
	N4 – et je sais rédiger un algorithme afin de répondre au problème posé.

Situation déclenchante de l'activité

Vous avez conçu un robot (séquence 10) et adapté une pièce pour déplacer un fût radioactif (séquence 11). Il vous reste à rédiger un algorithme pour que ce robot puisse :



- Suivre une ligne noire,
- Éviter un obstacle fixe (cube de 80 mm),
- Repérer et pousser un fût (cylindre de 40 mm),
- S'arrêter dans une zone de décontamination (boîte fermée).

L'agence vous fournit un algorithme partiel : votre rôle est de l'améliorer pour qu'il respecte les exigences du challenge.

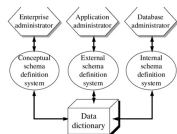
Mes constats, mes observations

Mon problème à résoudre

Mes idées pour le résoudre

Avant de rédiger un algorithme qui permettra au robot d'exécuter sa mission, tu dois d'abord comprendre ce qu'est un algorithme, à quoi il sert, et quelles sont les instructions de base qu'il peut contenir. Les questions suivantes te permettront de te familiariser avec ces notions indispensables à la suite du projet.

Activité N1.1 : Entourer l'image qui représente un algorithme



```

Début
Si (présence des élèves) alors
  Allumer les lumières
Sinon
  Éteindre les lumières
Fin
    
```

Activité N1.2 : A partir de la vidéo "Notions d'algorithmes", expliquer en quelques mots ce qu'est un algorithme.

Activité N2.1 : Répondre à la question suivante en choisissant parmi les propositions et expliquer votre choix

Dans quelle situation est-il le plus pertinent d'utiliser un algorithme ?

- a) Pour dessiner un paysage au hasard.
- b) Pour choisir sa couleur préférée.
- c) Pour trouver le chemin le plus court dans un labyrinthe.
- d) Pour décider de ce que l'on va manger au restaurant.

Activité N2.2 : Afin de rédiger un algorithme, on peut utiliser des instructions conditionnelles. A l'aide de la vidéo "Instructions conditionnelles", expliquer ce qu'est une instruction conditionnelle :

Activité N2.3 : A l'aide de la vidéo "Instructions itératives", expliquer ce qu'est une instruction itérative (une boucle) :

Ressources

Vidéo :



Notions d'algorithmes



Activités (niveaux 3 et 4)

Maintenant que tu sais ce qu'est un algorithme et comment il fonctionne, tu vas devoir analyser celui qui a été fourni par l'équipe de départ. Malheureusement, il ne permet pas encore au robot d'éviter correctement l'obstacle ni de gérer la récupération du fût radioactif. À toi de l'enrichir et de le corriger pour que le robot réalise la mission dans les conditions du challenge. Activité N3.1 : En reprenant la situation déclenchante, citer les éléments que le programmeur a oubliés afin de répondre au cahier des charges ? Formuler ces éléments à l'aide d'instructions conditionnelles et d'instructions itératives. Activité N4.1 : Rédiger l'algorithme complet en y intégrant les phrases manquantes. <i>Ton algorithme devra inclure la détection d'obstacle, la détection de faible luminosité, l'émission d'un signal, et l'arrêt dans la zone sécurisée.</i>	
--	--

Ma synthèse

	Fiches connaissances OST2b - les modes de représentation : croquis, schéma, graphique, algorithme, modélisation. SFC3c - instruction conditionnelle. SFC3d-Algorithmique et programmation : instructions itératives
--	--

Rappel des critères d'apprentissages de cette activité	N1 – Je sais reconnaître et expliquer ce qu'est un algorithme.
	N2 – et je sais justifier l'utilisation d'un algorithme
	N3 – et je sais modifier un algorithme permettant de répondre au besoin ou au problème posé.
	N4 – et je sais rédiger un algorithme afin de répondre au problème posé.

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

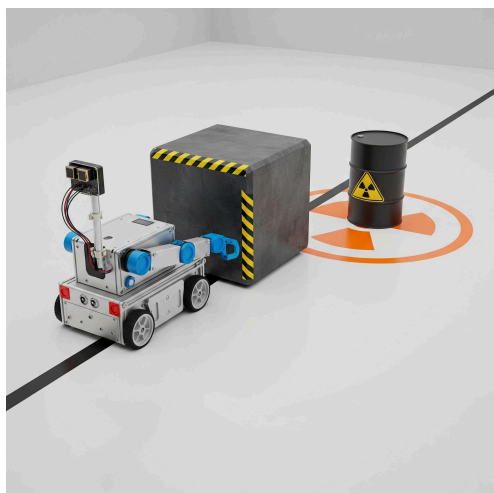
Classe :
Date :

Compétences et connaissances travaillées dans l'activité

Thème abordé : CCRI Attendu de fin de cycle : CCRI3-Concevoir, écrire, tester et mettre au point un programme Thématique : T22 - La programmation des OST	
Compétence CCRI31 - Modifier un algorithme permettant de répondre au besoin ou au problème posé.	Connaissances SFC3c - instruction conditionnelle. SFC3d-Algorithme et programmation : instructions itératives OST2b - les modes de représentation : croquis, schéma, graphique, algorithme, modélisation.
Critères d'apprentissages	N1 – Je sais reconnaître et expliquer ce qu'est un algorithme.
	N2 – et je sais justifier l'utilisation d'un algorithme
	N3 – et je sais modifier un algorithme permettant de répondre au besoin ou au problème posé.
	N4 – et je sais rédiger un algorithme afin de répondre au problème posé.

Situation déclenchante de l'activité

Vous avez conçu un robot (séquence 10) et adapté une pièce pour déplacer un fût radioactif (séquence 11). Il vous reste à rédiger un algorithme pour que ce robot puisse :



- Suivre une ligne noire,
- Éviter un obstacle fixe (cube de 80 mm),
- Repérer et pousser un fût (cylindre de 40 mm),
- S'arrêter dans une zone de décontamination (boîte fermée).

L'agence vous fournit un algorithme partiel : votre rôle est de l'améliorer pour qu'il respecte les exigences du challenge.

Mes constats, mes observations

L'algorithme est donné mais il semble incomplet par rapport au cahier des charges

Mon problème à résoudre

Comment modifier un algorithme permettant au robot de détecter un fût radioactif et de l'amener dans une zone de sécurité ?

Mes idées pour le résoudre

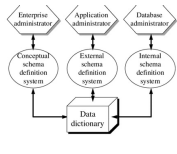
Activités (niveaux 1 et 2)

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

Activité N1.1 : Encadrer le schéma qui représente une suite d'instructions dans un ordre précis (Algorithmme)



Maintenant que tu sais ce qu'est un algorithme et comment il fonctionne, tu vas devoir analyser celui qui a été fourni par l'équipe de départ. Malheureusement, il ne permet pas encore au robot d'éviter correctement l'obstacle ni de gérer la récupération du fût radioactif. À toi de l'enrichir et de le corriger pour que le robot réalise la mission dans les conditions du challenge. Activité N3.1 : En reprenant la situation déclenchante, citer les éléments que le programmeur a oubliés afin de répondre au cahier des charges ? Formuler ces éléments à l'aide d'instructions conditionnelles et d'instructions itératives. SI le robot détecte un premier obstacle ALORS il tourne à gauche, il avance, il tourne à droite, il avance, il tourne à droite et il avance jusqu'à (répéter jusqu'à) retrouver la ligne SI le robot détecte un deuxième obstacle ALORS il avance en suivant la ligne et en poussant le fût jusqu'à ce que la luminosité diminue Activité N4.1 : Rédiger l'algorithme complet en y intégrant les phrases manquantes. Ton algorithme devra inclure la détection d'obstacle, la détection de faible luminosité, et l'arrêt dans la zone sécurisée.” Lorsque le mBot démarre, il suit la ligne noire. SI le mBot détecte un premier obstacle ALORS il tourne à gauche, il avance, il tourne à droite, il avance, il tourne à droite et il avance jusqu'à (répéter jusqu'à) retrouver la ligne SI le mBot détecte un deuxième obstacle ALORS il avance en suivant la ligne Lorsque le mBot détecte une luminosité plus faible, il s'arrête.	
---	--

Ma synthèse

Avant de pouvoir réaliser un programme, il faut d'abord rédiger un algorithme. Afin de rédiger correctement un algorithme et qu'il soit compréhensible de tous, nous pouvons utiliser des instructions conditionnelles ou itératives (boucles).	Fiches connaissances OST2b - les modes de représentation : croquis, schéma, graphique, algorithme, modélisation. SFC3c - instruction conditionnelle. SFC3d-Algorithme et programmation : instructions itératives
--	---

Rappel des critères d'apprentissages de cette activité	N1 – Je sais reconnaître et expliquer ce qu'est un algorithme.
	N2 – et je sais justifier l'utilisation d'un algorithme
	N3 – et je sais modifier un algorithme permettant de répondre au besoin ou au problème posé.
	N4 – et je sais rédiger un algorithme afin de répondre au problème posé.

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

Compétences et connaissances travaillées dans l'activité

Thème abordé : CCRI Attendu de fin de cycle : CCRI3-Concevoir, écrire, tester et mettre au point un programme Thématique : T22 - La programmation des OST	
Compétence CCRI32 - Traduire un algorithme permettant de répondre à un besoin ou à un problème simple en un programme.	Connaissances SFC3f - séquences (bloc) d'instructions. SFC3g - événement. SFC3h - déclenchement d'une séquence d'instructions par un événement. SFC3j - Programmation graphique par blocs. SFC3a - instruction d'affectation, variable (type mot, nombre et booléen)
Critères d'apprentissages	N1 – Je sais reconnaître et expliquer ce qu'est un programme.
	N2 – et je sais mettre en relation quelques capteurs ou actionneurs à différents blocs du logiciel
	N3 – et je sais traduire un algorithme permettant de répondre au besoin ou au problème posé en un programme
	N4 – et je sais réaliser un programme afin de répondre au problème posé.

Situation déclenchante de l'activité

Votre algorithme a été validé à l'issue de l'activité précédente. Vous avez conçu le robot mBot (séquence 10) et une pièce spéciale pour déplacer le fût (séquence 11). Il faut maintenant **programmer le robot à l'aide du logiciel mBlock pour réaliser sa mission dans le cadre du challenge robotique.



Mission à programmer :

- Le robot suit une ligne noire,
- Il contourne un obstacle fixe (cube 80x80x80 mm),
- Il repère un fût radioactif (cylindre Ø40 mm) et le pousse,
- Il atteint une zone de décontamination fermée (500x500x250 mm).

Votre défi : traduire l'algorithme précédent en blocs mBlock, en utilisant les capteurs et les événements.

Mes constats, mes observations

Mon problème à résoudre

Mes idées pour le résoudre

Activités (niveaux 1 et 2)

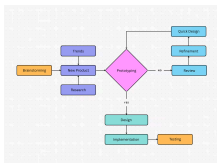
NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

Un algorithme seul ne suffit pas : pour que le robot fonctionne, il faut traduire cet algorithme en programme informatique. Avant cela, tu vas découvrir ce qu'est un programme, comment il se présente, et à quoi servent les différents blocs de programmation, en particulier ceux liés aux capteurs et actionneurs.

Activité N1.1 : Entourer l'image qui représente un programme.



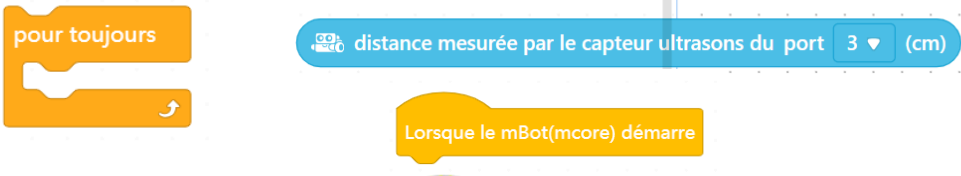
Activité N1.2 : A partir de la vidéo “Outils de description”, dire, par une phrase simple, ce que permet de faire un programme.

Activité N2.1 : Dans notre futur programme, le robot doit détecter des obstacles. A l'aide de la vidéo “Ecrire un programme”, répondre aux questions suivantes :

1) Entourer l'élément du robot qui permet de détecter un obstacle



2) Entourer le bloc du logiciel qui permet de calculer l'approche d'un obstacle



3) Pour que le capteur mesure l'approche de l'obstacle tout le long du circuit, quel bloc doit-on obligatoirement rajouter ?

Ressources
Vidéo :



Tu disposes maintenant d'un algorithme complet. Il est temps de passer à la phase de programmation. Tu vas utiliser les blocs de programmation dans mBlock pour que le robot suive la ligne, évite l'obstacle, détecte le fût radioactif et le transporte jusqu'à la zone de décontamination. Ces questions t'aideront à construire puis finaliser ton programme.

Activité N3.1 :
A partir de la vidéo “Notions de variables informatiques”, expliquer comment on peut compter les obstacles ?

Activité N3.2 :
Quels seront les blocs utilisés pour traduire l’algorithme ?
Compléter la partie du programme ci-dessous afin que le robot suive la ligne noire.



Activité N4.1 :
A partir des 2 questions précédentes, réaliser la programmation complète du mbot sur la plate-forme mBlock.
Teste ton programme. Si le robot atteint la zone de décontamination, appelle le professeur pour valider.

Ressources

[Notions de variables informatiques](#)



mBlock :
<https://ide.mblock.cc>

Ma synthèse

Fiches connaissances
[SFC3f - séquences \(bloc\) d'instructions.](#)
[SFC3g - événement.](#)
[SFC3h - déclenchement d'une séquence d'instructions par un événement.](#)
[SFC3a - instruction d'affectation, variable \(type mot, nombre et booléen\)](#)
[SFC3j - Programmation graphique par blocs.](#)

Rappel des critères d'apprentissages de cette activité	N1 – Je sais reconnaître et expliquer ce qu’est un algorithme.
	N2 – et je sais justifier l’utilisation d’un algorithme
	N3 – et je sais modifier un algorithme permettant de répondre au besoin ou au problème posé.
	N4 – et je sais rédiger un algorithme afin de répondre au problème posé.

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

Compétences et connaissances travaillées dans l'activité

Thème abordé : CCRI Attendu de fin de cycle : CCRI3-Concevoir, écrire, tester et mettre au point un programme Thématique : T22 - La programmation des OST	
Compétence CCRI32 - Traduire un algorithme permettant de répondre à un besoin ou à un problème simple en un programme.	Connaissances SFC3f - séquences (bloc) d'instructions. SFC3g - événement. SFC3h - déclenchement d'une séquence d'instructions par un évènement. SFC3j - Programmation graphique par blocs. SFC3a - instruction d'affectation, variable (type mot, nombre et booléen)
Critères d'apprentissages	N1 – Je sais reconnaître et expliquer ce qu'est un programme.
	N2 – et je sais mettre en relation quelques capteurs ou actionneurs à différents blocs du logiciel
	N3 – et je sais traduire un algorithme permettant de répondre au besoin ou au problème posé en un programme
	N4 – et je sais réaliser un programme afin de répondre au problème posé.

Situation déclenchante de l'activité

Votre algorithme a été validé à l'issue de l'activité précédente. Vous avez conçu le robot mBot (séquence 10) et une pièce spéciale pour déplacer le fût (séquence 11). Il faut maintenant **programmer le robot à l'aide du logiciel mBlock pour réaliser sa mission dans le cadre du challenge robotique.



Mission à programmer :

- Le robot suit une ligne noire,
- Il contourne un obstacle fixe (cube 80x80x80 mm),
- Il repère un fût radioactif (cylindre Ø40 mm) et le pousse,
- Il atteint une zone de décontamination fermée (500x500x250 mm).

Votre défi : traduire l'algorithme précédent en blocs mBlock, en utilisant les capteurs et les événements.

Mes constats, mes observations

Il y a un robot et un algorithme.

Mon problème à résoudre

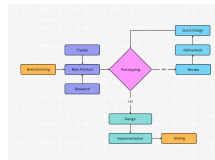
Comment traduire l'algorithme pour que le robot puisse réaliser les actions voulues ?

Mes idées pour le résoudre

Activités (niveaux 1 et 2)

Un algorithme seul ne suffit pas : pour que le robot fonctionne, il faut traduire cet algorithme en programme informatique. Avant cela, tu vas découvrir ce qu'est un programme, comment il se présente, et à quoi servent les différents blocs de programmation, en particulier ceux liés aux capteurs et actionneurs.

Activité N1.1 : Entourer l'image qui représente un programme



Activité N1.2 : A partir de la vidéo "Outils de description", dire, par une phrase simple, ce que permet de faire un programme.

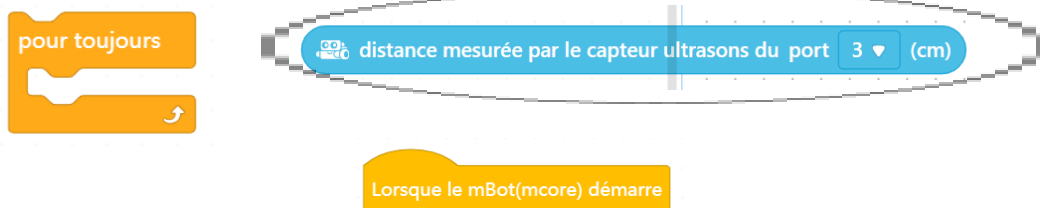
Un programme décrit le comportement d'un objet de façon exécutable par une machine, en utilisant un langage adapté.

Activité N2.1 : Dans notre futur programme, le robot doit détecter des obstacles. A l'aide de la vidéo "Ecrire un programme", répondre aux questions suivantes :

1) Entourer l'élément du robot qui permet de détecter un obstacle



2) Entourer le bloc du logiciel qui permet de calculer l'approche d'un obstacle



3) Pour que le capteur mesure l'approche de l'obstacle tout le long du circuit, quel bloc doit-on obligatoirement rajouter ?

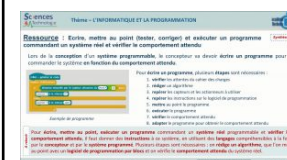
Il faut rajouter la boucle pour toujours

Ressources Vidéo :

Outils de description



Ecrire un programme



Tu disposes maintenant d'un algorithme complet. Il est temps de passer à la phase de programmation. Tu vas utiliser les blocs de programmation dans mBlock pour que le robot suive la ligne, évite l'obstacle, détecte le fût radioactif et le transporte jusqu'à la zone de décontamination. Ces questions t'aideront à construire puis finaliser ton programme.

Activité N3.1 :

A partir de la vidéo "Notion de variable informatique" expliquer comment on peut compter les obstacles ?

On utilise une variable "obstacle".
Mettre la variable "obstacle" à 0 au démarrage
Dès que le robot détecte un obstacle, il ajoute 1 à la variable "obstacle".
Si obstacle = 1 alors contourner obstacle
Si obstacle =2 alors avancer en suivant la ligne

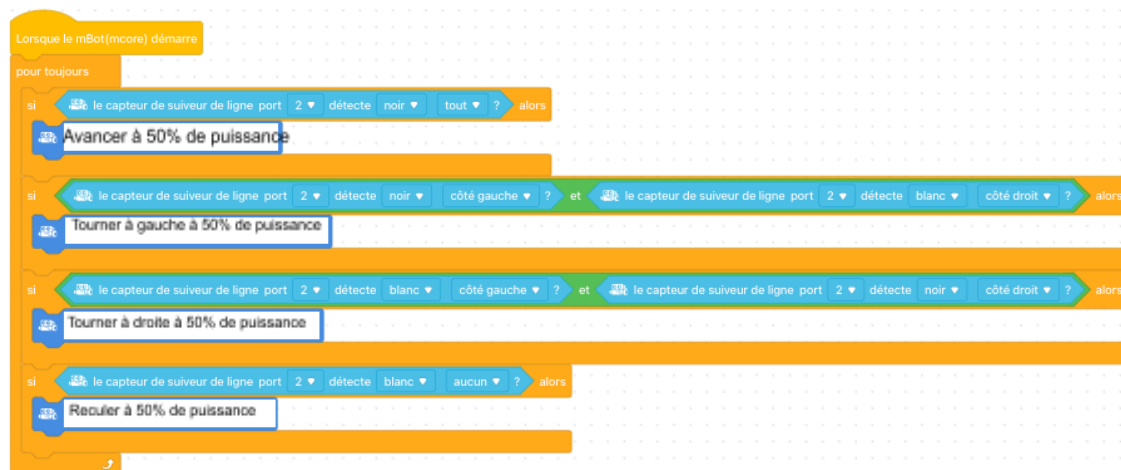
La variable "obstacle" permet donc de différencier les deux obstacles :

- obstacle = 1 → contourner,
- obstacle = 2 → suivre la ligne avec le fût.

Activité N3.2 :

Quels seront les blocs utilisés pour traduire l'algorithme ?

Compléter la partie du programme ci-dessous afin que le robot suive la ligne noire



Activité N4.1 :

A partir des 2 questions précédentes, réaliser la programmation complète du mbot sur la plate-forme mBlock.

Teste ton programme. Si le robot atteint la zone de décontamination, appelle le professeur pour valider.

Ressources



Notions de variables informatiques



mBlock :
<https://ide.mblock.cc>

Ma synthèse

Un programme est la traduction d'un algorithme sous forme de blocs exécutables. Il permet de piloter le robot mBot en fonction des capteurs et des événements. Pour réussir, on utilise des blocs conditionnels, des boucles et des variables.

Fiches connaissances

- [SFC3f - séquences \(bloc d'instructions\).](#)
- [SFC3g - événement.](#)
- [SFC3h - déclenchement d'une séquence d'instructions par un événement.](#)
- [SFC3a - instruction d'affectation, variable \(type mot, nombre et booléen\)](#)

NOM :
Prénom :

Groupe :
Rôle dans le groupe :

Classe :
Date :

	SFC3i - Programmation graphique par blocs.
--	--

Rappel des critères d'apprentissages de cette activité	N1 – Je sais reconnaître et expliquer ce qu'est un algorithme.
	N2 – et je sais justifier l'utilisation d'un algorithme
	N3 – et je sais modifier un algorithme permettant de répondre au besoin ou au problème posé.
	N4 – et je sais rédiger un algorithme afin de répondre au problème posé.

 Région académique NOUVELLE-AQUITAINE	STRUCTURE, FONCTIONNEMENT, COMPORTEMENT : des objets et systèmes techniques à comprendre	 <i>Cycle 4</i>
	La programmation d'une nouvelle fonctionnalité	
Connaissance	Algorithmique et programmation : variable	

En informatique, **une variable est un élément fondamental qui permet de stocker et de manipuler des données**. Les programmeurs utilisent des **variables informatiques** dans leurs programmes **pour permettre aux objets et systèmes techniques de fonctionner**.

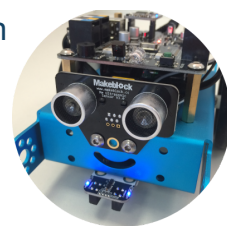
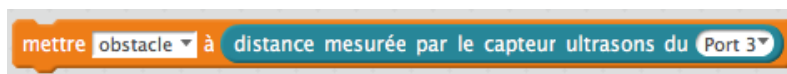
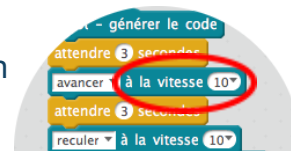
La variable se compose de **deux éléments essentiels** :

- **Le nom** : Il s'agit d'un **identifiant unique** qui permet de référencer la variable dans le programme.
Le choix du nom doit être clair et explicite pour refléter la nature des données stockées.
- **La valeur** : C'est la **donnée concrète** stockée dans la variable.
La nature de la valeur dépend du type de variable, qui peut être :
 - o **Entier**: Nombres entiers, positifs, négatifs ou nuls (ex: 1, -10, 0)
 - o **Réel**: Nombres décimaux (ex: 3.14, -5.2, 2.71828)
 - o **Chaîne de caractères**: Séquences de caractères (ex: "Bonjour", "123 Main Street", "Python")
 - o **Booléen**: Valeurs booléennes "vrai" ou "faux" (ex: True, False)



L'utilisation des variables :

- Elles peuvent être utilisées pour définir une **valeur** précise à un **actionneur** (ex: vitesse d'un moteur)..
- Elles peuvent prendre la **valeur** mesurée par un **capteur** (ex : **distance** d'un **obstacle**).



- La plupart du temps, elles sont **prédéfinies** dans le **logiciel de programmation**.



Une **variable** peut être utilisée pour **définir une valeur** précise à un **actionneur** ou **contenir une valeur** issue d'un **capteur**. Ces **valeurs** peuvent **évoluer** au cours de l'exécution du programme (exemple : une distance). Elles sont stockées à l'intérieur de la **mémoire** du système.

 Région académique NOUVELLE-AQUITAINE	STRUCTURE, FONCTIONNEMENT, COMPORTEMENT : des objets et systèmes techniques à comprendre	 <i>Cycle 4</i>
	La programmation d'une nouvelle fonctionnalité	
Connaissance	Algorithmique et programmation : déclenchement d'une séquence d'instructions par un événement	

Pour rendre un programme réactif à son environnement et aux interactions avec l'utilisateur, on met en place des **déclenchements de séquences d'instructions par des événements**.

Un **événement** est un **changement** qui **survient** dans l'environnement d'un objet ou d'un **système**.

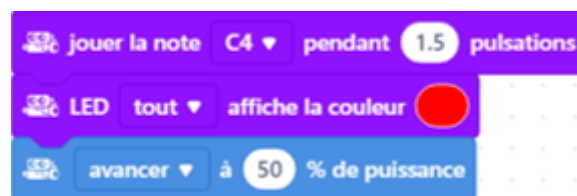
Il peut s'agir :

- d'une **action humaine** (clic sur un bouton)
- d'un **phénomène naturel** (détection de lumière)
- d'une **information reçue** (signal d'un capteur)

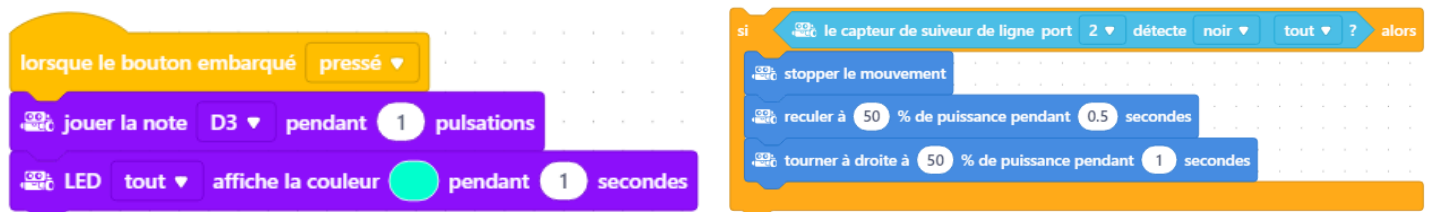


Une **séquence d'instructions** est un **ensemble d'ordres** donnés à un objet ou à un système pour qu'il effectue une tâche.

Ici, on demande à un système programmable de jouer une note, puis d'allumer ses LED en rouge, puis d'avancer à 50% de sa puissance.



Le **déclenchement d'une séquence d'instructions par un événement** est le principe qui permet de rendre les objets et les systèmes techniques interactifs.



Les **événements (actions, phénomènes, informations)** permettent de **déclencher des séquences d'instructions** dans un programme, **rendant l'objet ou le système interactif**. Un événement peut être une **action humaine**, un **phénomène naturel** ou une **information reçue**. La **séquence d'instructions** définit les **actions à effectuer en réponse à l'événement** (ex: jouer une note, allumer une LED, avancer).

 Région académique NOUVELLE-AQUITAINE	STRUCTURE, FONCTIONNEMENT, COMPORTEMENT : des objets et systèmes techniques à comprendre	 <i>Cycle 4</i>
	La programmation d'une nouvelle fonctionnalité	
Connaissance	Programmation graphique par blocs	

Lorsqu'on veut donner des instructions à un robot ou à un jeu vidéo, on peut utiliser la **programmation graphique par blocs**. C'est un moyen **simple et visuel** de **créer des programmes**, sans avoir à écrire du code.

La programmation par blocs **fonctionne avec des blocs de code qui s'emboîtent les uns dans les autres**. Chaque bloc **représente une action ou une condition**. En **assemblant les blocs**, on crée un programme qui **définit le comportement de l'objet ou du système technique**.



Il existe différents types de blocs :

• Blocs d'action

Définissent les **actions que l'objet ou le système doit réaliser** (ex: avancer, tourner, afficher un message).



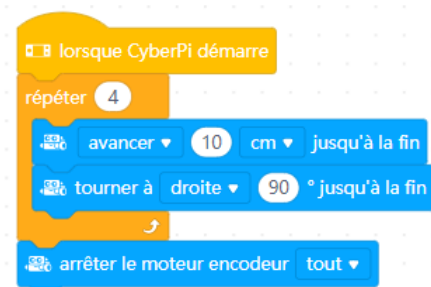
• Blocs de boucle

Définissent la **répétition** d'une série d'actions (ex: répéter 4 fois avancer et tourner).



• Blocs de condition

Définissent les **conditions qui doivent être remplies** pour que les actions soient exécutées (ex: si l'objet est proche d'un obstacle, alors s'arrêter).



Pour programmer par bloc, il faut savoir

- Identifier les différents types de blocs.
- Assembler les blocs pour créer un programme simple.
- Tester un programme.
- Modifier les paramètres d'un programme et identifier ses effets.

La **programmation par blocs** est un moyen simple et visuel de créer des programmes sans écrire de code. Elle utilise des **blocs de code emboîtables** qui représentent **des actions, des conditions ou des boucles**. Pour programmer par blocs, il faut **identifier les types de blocs, les assembler, tester le programme et modifier ses paramètres**.

	EVALUATION DE COMPÉTENCE	
NOM : Prénom :	Classe : Date :	

Observations et conseils de progrès

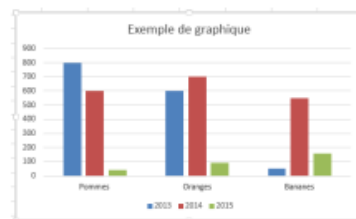
Référence aux compétences et connaissances évaluées

Thème abordé : Attendu de fin de cycle : CCRI3 - Concevoir, écrire, tester et mettre au point un programme Thématique : T22 - La programmation des OST	
Compétence détaillée CCRI31 - Modifier un algorithme permettant de répondre au besoin ou au problème posé. Compétence détaillée CCRI32 - Traduire un algorithme permettant de répondre à un besoin ou à un problème simple en un programme.	Connaissance SFC3f - séquences (bloc) d’instructions. SFC3g - événement. SFC3h - déclenchement d’une séquence d’instructions par un évènement. SFC3j - Programmation graphique par blocs. SFC3a - instruction d’affectation, variable (type mot, nombre et booléen) SFC3c - instruction conditionnelle. SFC3b - opérateurs arithmétiques et logiques (ET, OU, NON). SFC3d - Algorithmique et programmation : instructions itératives OST2b - les modes de représentation : croquis, schéma, graphique, algorithme, modélisation.
Critères d'apprentissages	N1 – Je sais reconnaître un algorithme et identifier quelques éléments basiques d'un algorithme
	N2 – et je sais concevoir ou modifier des algorithmes simples (avec conditions, variables)
	N3 – et je sais traduire des algorithmes simples en programme
	N4 – et je sais réaliser ou mettre au point des programmes

N1 – Je sais reconnaître un algorithme

N1a – Entourer l’image qui montre un algorithme :

- Le robot avance de 5m
- Le robot tourne à gauche de 30°
- Le robot avance de 3m
- Le robot tourne à gauche de 60°
- Le robot avance de 2m
- Le robot tourne à gauche de 90°
- Le robot avance de 7,6m
- Le robot tourne à gauche de 90°
- Le robot avance de 3,5m



N1b – Relier à la bonne définition

Un algorithme est ●

- une représentation simplifiée, symbolique et normalisée d'un objet ou d'un système
- une suite d'instructions ordonnées et précises qui permet de réaliser une tâche
- un dessin rapide et simplifié réalisé à main levée

N2 – Je sais concevoir ou modifier des algorithmes simples (avec conditions, variables)

N2a - Rédige l'algorithme permettant, dès la détection d'une personne , l'ouverture de la porte du magasin pendant 5 secondes.



N2b – Modifie cet algorithme afin qu'il tienne compte d'un obstacle à éviter (à l'aide d'un capteur), puis continue sa route pour pousser un fût vers une boîte de décontamination.

N3 & N4 – Je sais traduire des algorithmes simples en programme et je sais réaliser ou mettre au point des programmes

On souhaite programmer un robot afin qu'il contourne un obstacle.

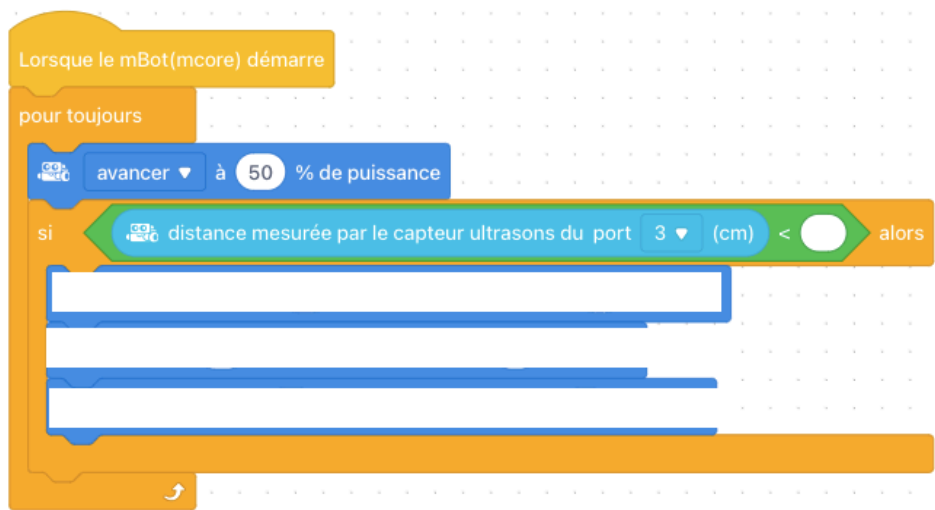
L'algorithme est fourni :

Lorsque le mBot démarre, il avance.

Si le mBot détecte un obstacle à moins de 10 cm ALORS il tourne à gauche pendant 1 seconde, il avance pendant 1 seconde, il tourne à droite pendant 1 seconde

Si il y a encore un obstacle, il recommence cette manoeuvre SINON il avance

Compléter le programme ci-dessous pour qu'il corresponde à l'algorithme à l'aide des propositions suivantes : Suivre la ligne noire - 10cm - Avancer pendant 1s - Tourner à droite pendant 1s - Reculer pendant 2s - Tourner à gauche pendant 1s - 5cm



	EVALUATION DE COMPÉTENCE	
NOM : Prénom :	Classe : Date :	

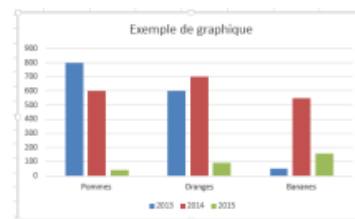
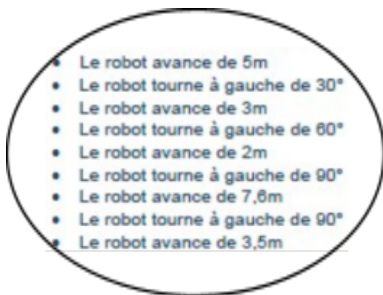
Observations et conseils de progrès

Référence aux compétences et connaissances évaluées

Thème abordé : Attendu de fin de cycle : CCRI3 - Concevoir, écrire, tester et mettre au point un programme Thématique : T22 - La programmation des OST	
Compétence détaillée CCRI31 - Modifier un algorithme permettant de répondre au besoin ou au problème posé. Compétence détaillée CCRI32 - Traduire un algorithme permettant de répondre à un besoin ou à un problème simple en un programme.	Connaissance SFC3f - séquences (bloc) d'instructions. SFC3g - événement. SFC3h - déclenchement d'une séquence d'instructions par un événement. SFC3j - Programmation graphique par blocs. SFC3a - instruction d'affectation, variable (type mot, nombre et booléen) SFC3c - instruction conditionnelle. SFC3b - opérateurs arithmétiques et logiques (ET, OU, NON). SFC3d - Algorithmique et programmation : instructions itératives OST2b - les modes de représentation : croquis, schéma, graphique, algorithme, modélisation.
Critères d'apprentissages	N1 – Je sais reconnaître un algorithme et identifier quelques éléments basiques d'un algorithme
	N2 – et je sais concevoir ou modifier des algorithmes simples (avec conditions, variables)
	N3 – et je sais traduire des algorithmes simples en programme
	N4 – et je sais réaliser ou mettre au point des programmes

N1 – Je sais reconnaître un algorithme

N1a – Entourer l’image qui montre un algorithme :



N1b – Relier à la bonne définition

- une représentation simplifiée, symbolique et normalisée d'un objet ou d'un système

Un algorithme est



- une suite d'instructions ordonnées et précises qui permet de réaliser une tâche
- un dessin rapide et simplifié réalisé à main levée

N2 – Je sais concevoir ou modifier des algorithmes simples (avec conditions, variables)

N2a - Rédige l'algorithme permettant, dès la détection d'une personne , l'ouverture de la porte d'entrée du magasin pendant 5 secondes.



Si une personne est détectée alors ouvrir la porte pendant 5s.

Refermer la porte

N2b – Modifie cet algorithme afin qu'il tienne compte d'un obstacle à éviter (à l'aide d'un capteur), puis continue sa route pour pousser un fût vers une boîte de décontamination.

Capteur 1 : capteur ouverture

Capteur 2 : capteur fermeture

Si une personne est détectée par le capteur 1 alors ouvrir la porte.

Si une personne est détectée par le capteur 2 alors fermer la porte.

N3 & N4 – Je sais traduire des algorithmes simples en programme et je sais réaliser ou mettre au point des programmes

On souhaite programmer un robot afin qu'il contourne un obstacle.

L'algorithme est fourni :

Lorsque le mBot démarre, il avance.

Si le mBot détecte un obstacle à moins de 10 cm ALORS il tourne à gauche pendant 1 seconde, il avance pendant 1 seconde, il tourne à droite pendant 1 seconde

Si il y a encore un obstacle, il recommence cette manoeuvre SINON il avance

Compléter le programme ci-dessous pour qu'il corresponde à l'algorithme à l'aide des propositions suivantes : Suivre la ligne noire - 10 - Avancer pendant 1s - Tourner à droite pendant 1s - Reculer pendant 2s - Tourner à gauche pendant 1s- 5

