

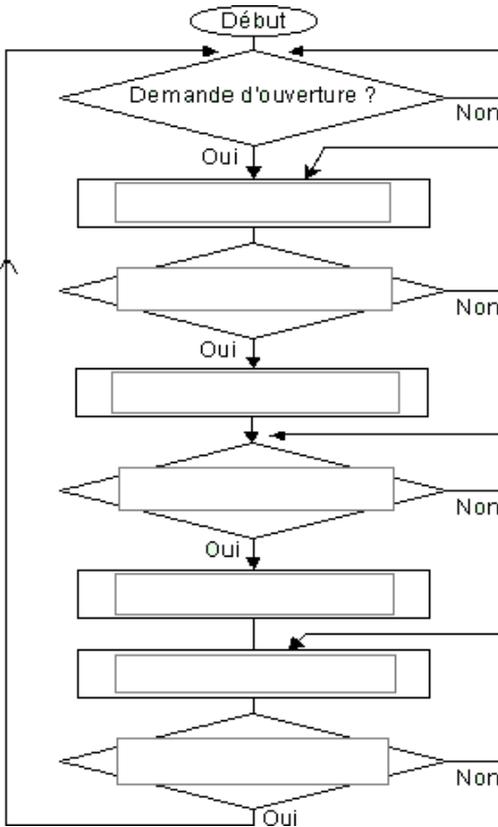
Ecrire un algorithme ou un programme qui permet une interaction avec l'utilisateur

BARRIERE DE PARKING ET ESCALATOR

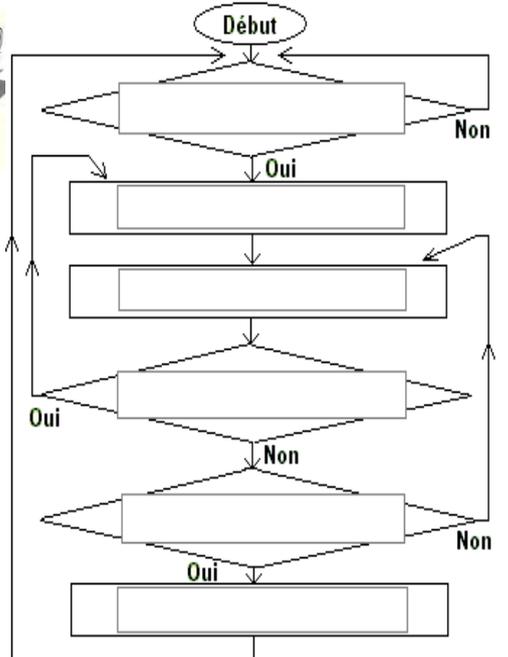
1. Réaliser les exercices en ligne et recopier vos réponses



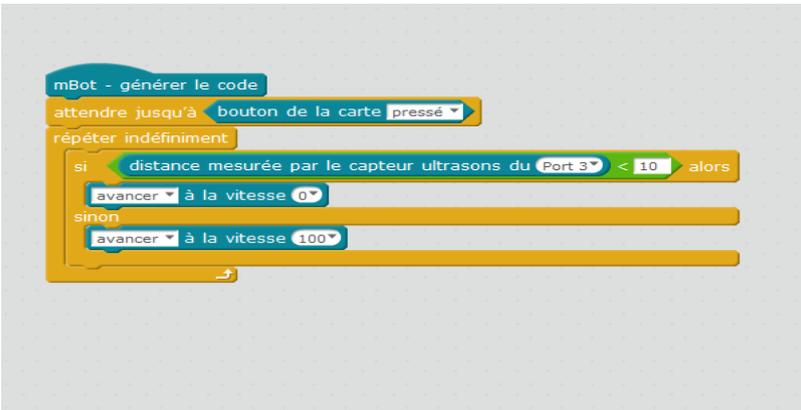
Trouvez l'organigramme sachant que le feu doit passer au vert lorsque la barrière est ouverte, et repasse au rouge juste avant de se baisser.



Trouvez l'organigramme de l'escalator pour qu'il se mette en route pendant 10 secondes quand quelqu'un se présente en bas. Si une autre personne se présente avant la fin des 10 secondes, il faut remettre à zéro la minuterie



ROBOT livreur Mbot

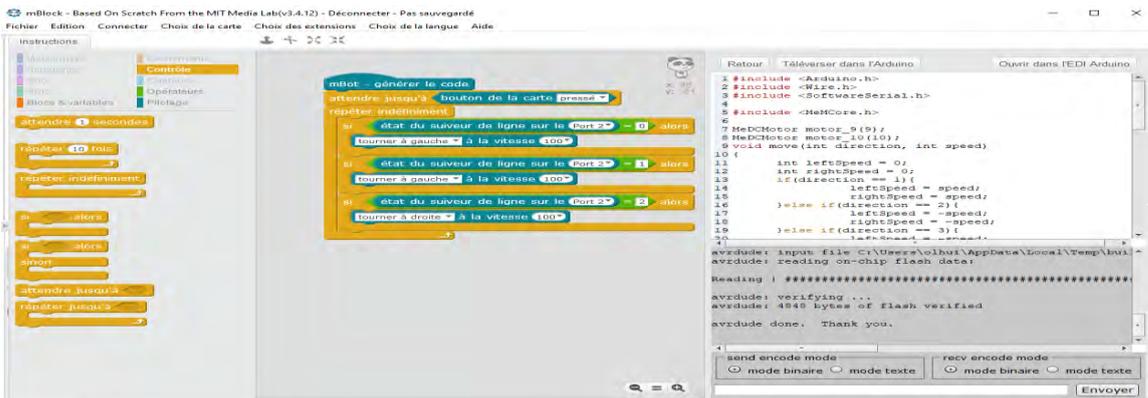


Ouvrir le programme Mblock présent sur l'ordinateur . Tester le programme arrêt si obstacles et le Tuto vidéo pour Le suiveur de ligne.

Ensuite Créer un programme pour que **Le Mbot** suive le trajet de livraison et s'arrêter si il rencontre un Obstacle.

Pour Téléverser votre Programme, connecter votre Mbot Sur le 2eme port COM

Dans Edition : Choisissez cacher la scène et mode Arduino

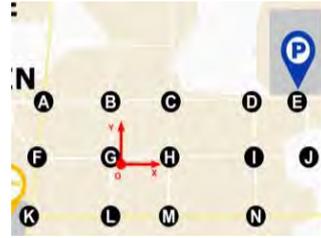


Ecrire un algorithme ou un programme qui permet une interaction avec l'utilisateur

Réaliser le programme permettant de se rendre au point N et revenir au magasin

Réaliser le programme permettant de se rendre au point B en passant par L et revenir au magasin " noter le sur votre feuille de cours"

Réaliser le programme (Point bonus) permettant de se rendre au point D en passant par G et revenir au magasin



	X	Y
O	0	0
A	-90	80
B	-10	80
C	60	80
D	155	80
E	210	80

lorsque vous cliquez sur

allez à x: y:

pointer en direction de

glisser secondes vers x: y:

pointer en direction de

glisser secondes vers x: y:

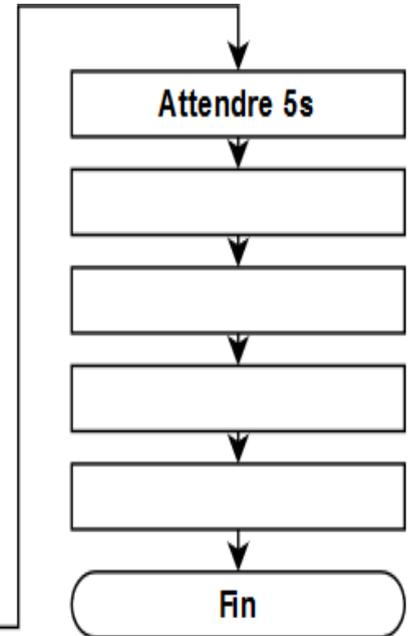
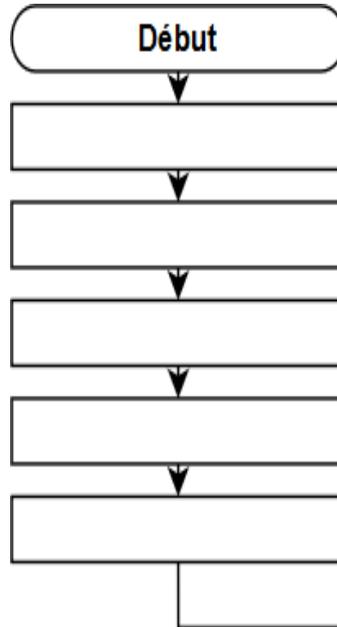
attendre 5 secs

pointer en direction de

glisser secondes vers x: y:

pointer en direction de

glisser secondes vers x: y:



ROBOT LIVREUR Thymio

Réaliser les programmes proposés et observer le comportement de Thymio, trouver une solution pour obtenir le résultat décrit dans l'Objectif en vidéo, compléter le programme ci- dessous pour aller du magasin au client

quand un bouton flèche est appuyé

si le bouton flèche avant est appuyé

faire

- allume toutes les lumières couleur
- tant que non
- faire
 - si et
 - faire vitesse des roues à gauche % à droite %
 - si et
 - faire vitesse des roues à gauche % à droite %
 - si et
 - faire vitesse des roues à gauche % à droite %

arrête

Ecrire un algorithme ou un programme qui permet une interaction avec l'utilisateur

Portes automatiques de magasin :

Objectif : les portes s'ouvrent lorsque le breakdancer s'approche

Compléter et tester les programmes

```
lorsque vous cliquez sur [ ]
pour toujours
  si [ distance vers Breakdancer1 < 0 ] alors
    définir Détection à 0
  sinon
    définir Détection à 0
```

```
lorsque vous cliquez sur [ ]
pour toujours
  si [ touche flèche de droite pressée ? ] alors
    modifier x de 10
  si [ touche flèche de gauche pressée ? ] alors
    modifier x de -10
```

Portes automatiques de magasin :

Objectif: La lampe s'éteint si le seuil de luminosité est supérieur à 80

Compléter et tester les programmes

```
lorsque vous cliquez sur [ ]
pour toujours
  définir Luminosité à Luminosité simulée
  si [ Luminosité > 80 ] alors
    basculer le costume sur abribus-lampe-off
```

Écrire un algorithme ou un programme qui permet une interaction avec l'utilisateur

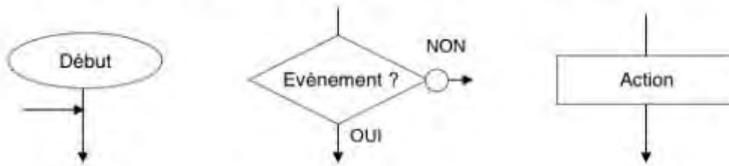
Symboles de base



Un algorithme est une suite d'instructions précises et structurées qui décrit la manière dont on résout un problème.

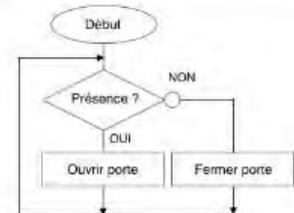
Cette description peut être textuelle (si, alors, sinon, tant que ...) ou graphique (appelé également organigramme ou logigramme).

Dans ce cas des normes d'écritures sont à respecter :



```

Début
Si Présence
  Alors ouvrir porte
  Sinon fermer porte
Fin Si
Retour au début
    
```



Boucles



Lorsque des instructions sont répétées, on utilise des boucles pour optimiser le programme.

Exemples de boucles :

Répéter indéfiniment, Répéter x fois, Répéter jusqu'à ...



Arduino - générer le code

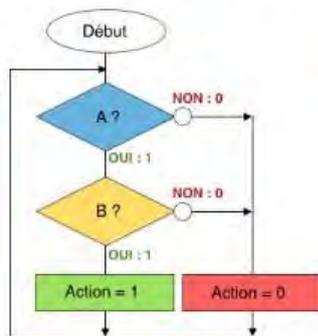


Fonctions ET et OU

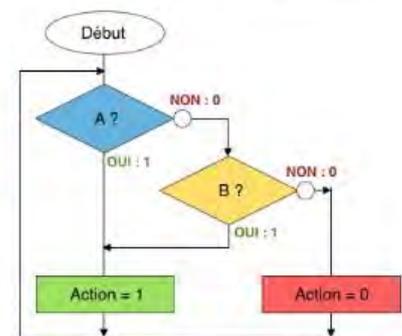
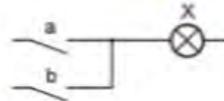
L'utilisation des fonctions ET et OU sont essentielles pour présenter correctement une solution.



A ?	B ?	Sortie
0	0	0
0	1	0
1	0	0
1	1	1



A ?	B ?	Sortie
0	0	0
0	1	1
1	0	1
1	1	1

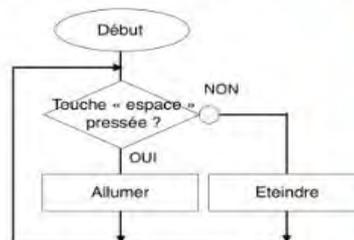


Déclenchement d'une action par un événement, instructions conditionnelles



```

Début
Si touche « espace » pressée
  Alors allumer
  Sinon éteindre
Fin Si
Retour début
    
```



Variable informatique



Une variable est une donnée (une information) associée à un nom. Elle est mémorisée/stockée et elle peut changer de valeur en fonction des instructions du programme.

Exemple : variable « compteur »

