

Compétences :

- Ecrire et mettre au point (tester, corriger) et exécuter un programme commandant un système réel et vérifier le comportement attendu.
- Ecrire un programme dans lequel des actions sont déclenchées par des événements extérieurs.



✓ Les objets connectés sont souvent programmés pour fonctionner automatiquement. La résolution d'un problème par un programmeur peut s'effectuer en trois étapes :

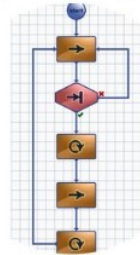
-1- Ecriture d'un algorithme : Suites logique d'opérations ou d'instructions, souvent rédigées sur feuille de papier en utilisant le langage naturel et des mots clés : si, alors, tant que, jusqu'à ...

Exemple : Un robot évitant un obstacle.

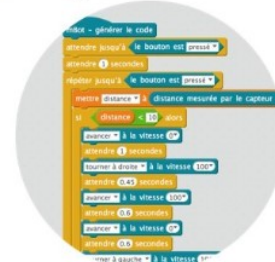


~ Si le robot détecte un obstacle avec son capteur de pare-choc, alors tourner à gauche de 90°, avancer de 10cm puis tourner à droite de 90°.
~ Sinon avancer indéfiniment.

-2- Construction à l'aide d'un logiciel d'une représentation graphique de l'algorithme



Algorithme : organigramme de programmation



Logiciel de représentation graphique par bloc (ou briques) comme Scratch

-3- Traduction de la représentation graphique en langage de programmation qui lui-même sera converti en langage machine (code binaire) que le microprocesseur peut exécuter

```

17 // ...
18 [
19   distance = ultrasonic_3.distanceCm();
20   if (distance < 10) {
21     motor.move(1, 0);
22     delay(1000*0.45);
23     motor.move(1, 100);
24     delay(1000*0.6);
25     motor.move(1, 0);
26     delay(1000*0.6);
27     motor.move(1, 100);
28     delay(1000*0.45);
29     motor.move(1, 100);
30   } else {
31     motor.move(1, 100);
32   }
33 ]
    
```

Programme : lignes de codes en langage C

Pour résoudre un problème, le programmeur commence par écrire un algorithme dans lequel il donne des ordres en fonction de conditions (état des capteurs). Puis il construit sur un ordinateur une représentation graphique de l'algorithme (Algorithme ou par bloc avec Scratch). Le logiciel va ensuite traduire la représentation graphique en ligne de code (le programme) que le système va exécuter.

✓ Dans les algorithmes, on peut utiliser des boucles, des instructions conditionnelles, et faire des calculs avec des variables informatiques. On peut aussi créer des sous-fonctions.

LES BOUCLES :

on répète un certain nombre de fois une même instruction.

Exemple 1 :

On indique au début de la boucle le nombre de répétitions souhaitées.

On indique dans la boucle l'instruction qu'on souhaite répéter.

Exemple 2 :

On indique au début de la boucle qu'on la répète indéfiniment.

On indique dans la boucle l'ensemble des instructions qu'on souhaite répéter.

LES INSTRUCTIONS CONDITIONNELLES :

on réalise certaines instructions si une condition précise est vérifiée.

Exemple 1 :

On indique la condition à vérifier.

On indique l'instruction à suivre si la condition est vraie.

Exemple 2 :

On indique l'instruction à suivre si la condition est vraie.

On indique l'instruction à suivre si la condition est fausse.

LES SOUS-FONCTIONS :

On crée une sous-fonction dans « Créer un bloc » à qui l'on donne un nom, par exemple « dessiner un carré »

On définit ce que doit faire la sous-fonction :

Puis, dans le programme principal, on peut appeler la sous-fonction autant de fois qu'on le souhaite. À chaque appel, toutes ses instructions seront réalisées.

LES VARIABLES :

On crée un nom de variable, par exemple X, qui sera associé à une valeur pouvant varier durant l'exécution du programme.

On peut donner une valeur à la variable (ici X = 0)



On peut augmenter la valeur de la variable (ici X augmente de 1)



On peut vérifier des conditions sur la variable (ici X < 10).



On peut faire des opérations avec la variable (ici on l'additionne avec Y)



