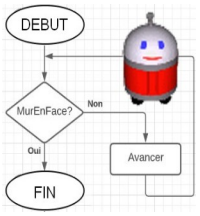
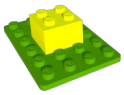
Ce que je dois retenir

ORGANIGRAMME
ALGORIGRAMME OU LOGIGRAMME



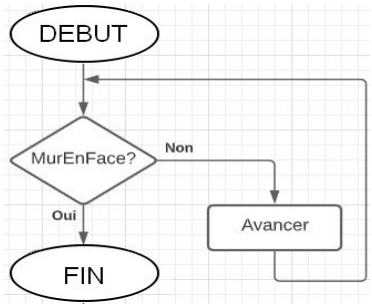
CT 1.3-CT 2.5-CT 2.7-DIC 1.5 CT 3.1-OTSCIS 2.1 CT 4.2-CT 5.5-IP 2.3	Imaginer des solutions pour produire des éléments de programmes informatiques en réponse au besoin. Exprimer sa pensée à l'aide d'outils de description adaptés : croquis, schémas, graphes, diagrammes, tableaux. Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs.

Qu'est ce qu'un organigramme (ou logigramme)



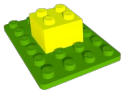
Un algorigramme est une représentation sous forme d'un schéma d'un programme informatique.

Exemple d'organigramme :



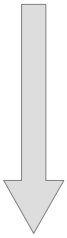
Organigramme
=
Algorigramme
=
Logigramme

Règles pour faire un Organigramme



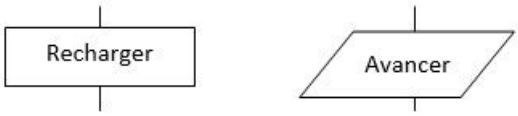
Un organigramme doit respecter des normes d'écritures :

- Il commence toujours par la case Début
- Il s'écrit de haut en bas
- Il finit presque toujours une case Fin



Les cases d'actions :

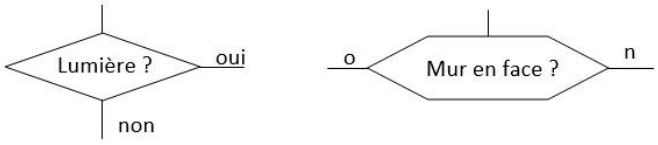
- 1 entrée (toujours au-dessus)
- 1 sortie (toujours en-dessous)
- dans la case il y a un **verbe**



2 représentations possibles

Les cases de tests :

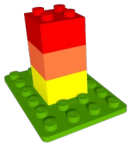
- 1 entrée (toujours au-dessus)
- 2 sorties
- dans la case il y a une **question**



2 représentations possibles

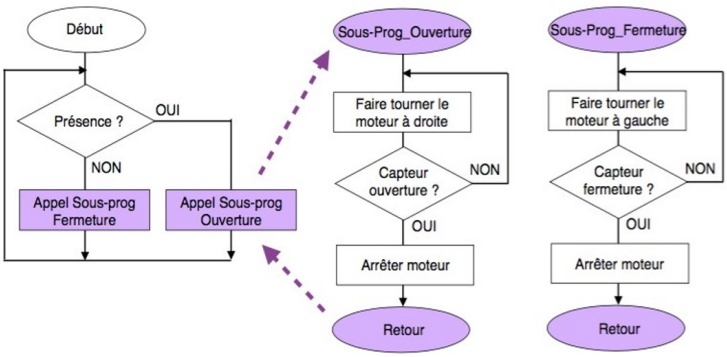
Savoir dessiner une case test et une case action

Algorithme et gestion des sous-problèmes

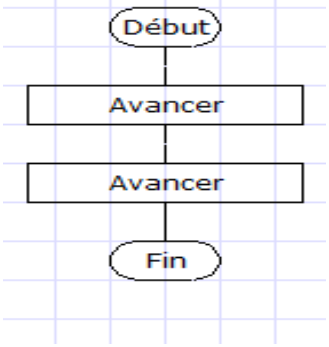


Pour rendre le programme plus court, il est parfois utile de décomposer les problèmes en sous-problèmes.

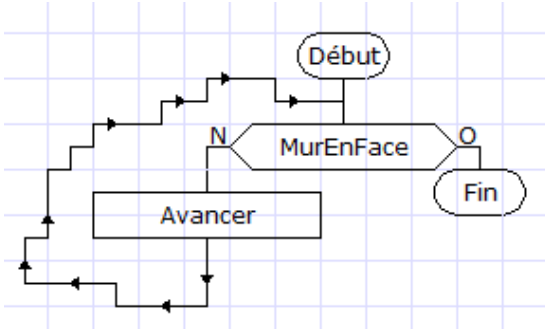
On fait alors des **sous-programmes** qui seront **appelés par le programme principal**.



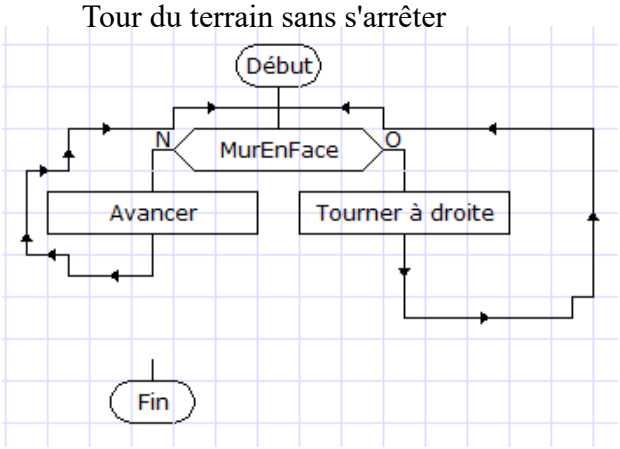
Programme
Avancer de 2 cases



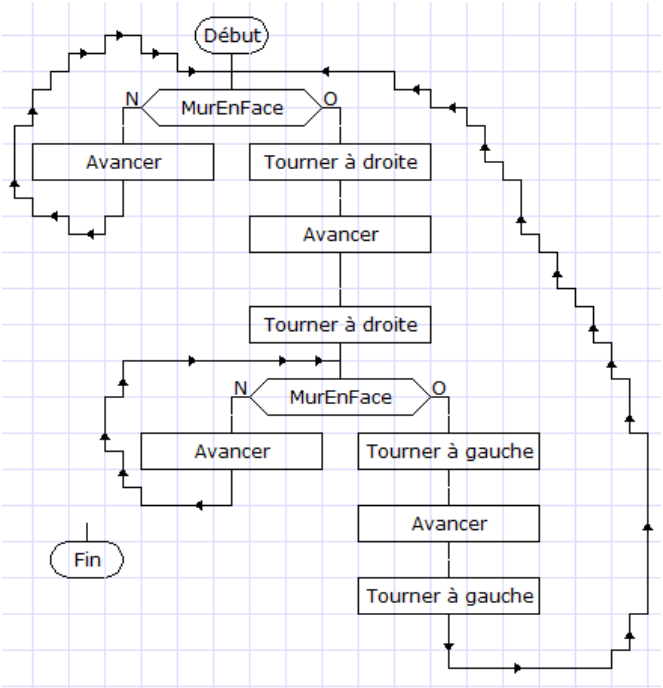
Programme : Mur en face



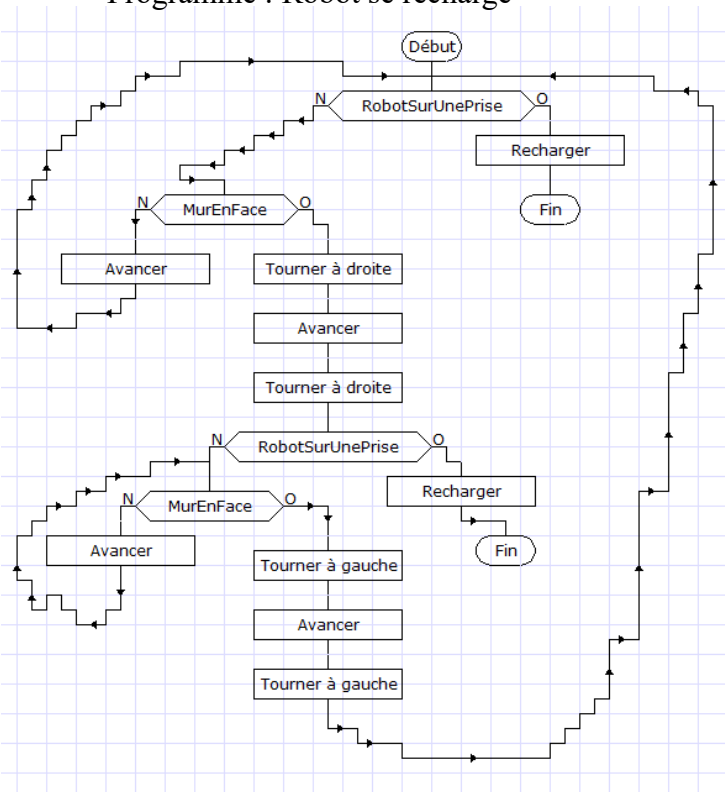
Programme :
Tour du terrain sans s'arrêter



Programme : Zig-zag



Programme : Robot se recharge



correction

Savoir expliquer ces programmes et les reconnaitre.