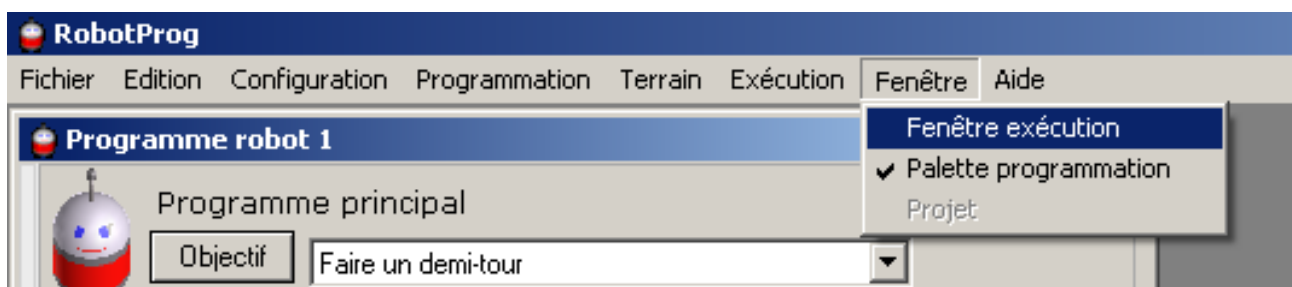




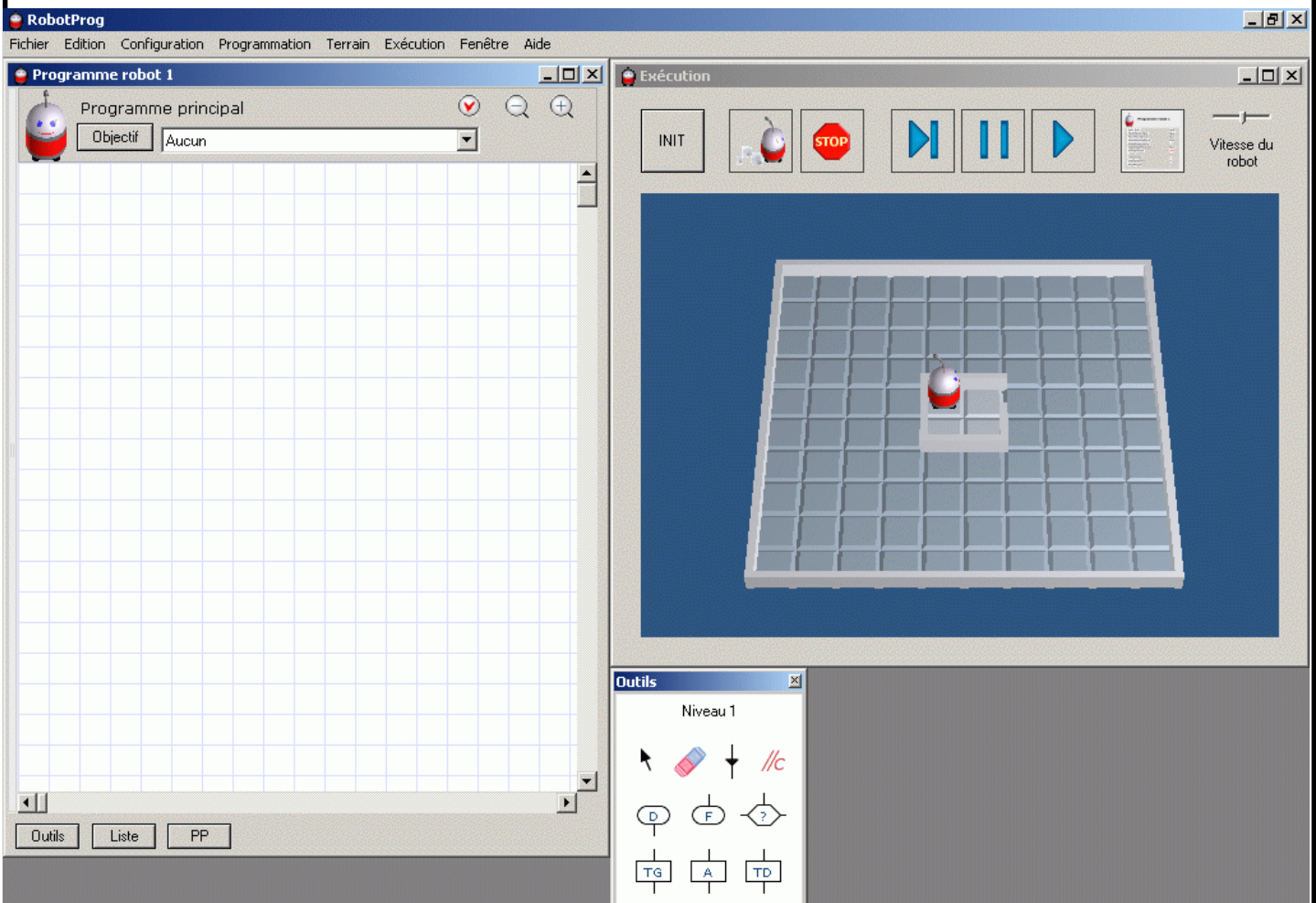
Ce que j'apprends : Dans cette activité, tu va programmer un robot virtuel. L'objectif est de maîtriser la programmation par organigramme et de pouvoir réaliser des tâches simples quelque défis simple. Tu doit être capable d'expliquer et de modifier un organigramme pour en améliorer le fonctionnement

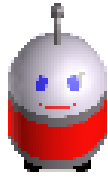
I Lancement et configuration du programme :

- 1- Clique deux fois sur l'icône pour lancer le programme **Robotprog**.
- 2- Dans le menu **Fenêtre**, sélectionne **Fenêtre exécution** afin de faire apparaître le terrain d'évolution du robot.



- 3- Diminue au maximum la taille de la fenêtre Exécution et fais en sorte que ton écran ait l'apparence suivante :





II Prise en main du logiciel : Écrire le programme

Pour programmer ton robot, tu disposes de la fenêtre **Outils** qui contient tous les blocs nécessaires à la réalisation de tes organigrammes.

- Pour utiliser un bloc, il suffit de cliquer dessus, puis de cliquer dans la fenêtre **Programme** à l'endroit où tu veux le placer.
- pour lier les blocs entre eux, choisissez l'outil lien dans la palette ; cliquez sur une sortie de bloc, déplacez la souris et cliquez sur l'entrée du bloc suivant.
- Pour effacer, il suffit de cliquer sur la **gomme**, puis sur le bloc à effacer.

Fenêtre programme

Palette d'outils

Tracer un lien

Création de blocs

Programme principal
Objectif : Aucun

Début

Avancer

Tourner à droite

Avancer

Fin

Outils

Liste

PP

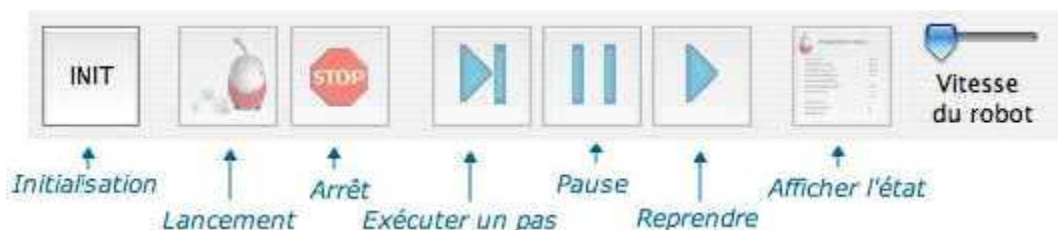
Un organigramme doit contenir un bloc début et un seul pour indiquer où le programme doit commencer, et un ou plusieurs blocs fin.

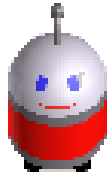
D : Début
F : Fin
TG : Tourne à gauche
TD : Tourne à droite
A : Avance

Initialisation du programme : cliquez sur le bouton INIT Le programme est alors vérifié. L'initialisation permet de mettre le programme à l'état « Début »

Si le programme contient une erreur, vous ne pourrez pas lancer l'exécution, vous devez d'abord corriger l'erreur.

Lancement du programme : cliquez sur le bouton Lancement





II Prise en main du logiciel : Les déplacements du robot

Pour déplacer le robot, vous disposez de trois commandes :

Avancer, Tourner à droite, Tourner à gauche.

Ces commandes sont écrites dans des blocs de forme rectangulaire

* **La commande Avancer** fait avancer le robot d'une case devant lui. Attention, si le robot est en face d'un mur quand cette commande est exécutée, il s'écrase contre le mur, c'est une erreur d'exécution, le programme s'arrête.

* **Les commandes Tourner à droite et Tourner à gauche** font faire un quart de tour au robot vers sa droite ou sa gauche. Le robot reste dans la même case.

III Mon premier programme :

Fermez le programme précédent et choisissez le menu Fichier > Nouveau programme.

Écrire un programme faisant faire un demi-tour au robot.

Exécuter ce programme et recopier votre organigramme sur une feuille de copie

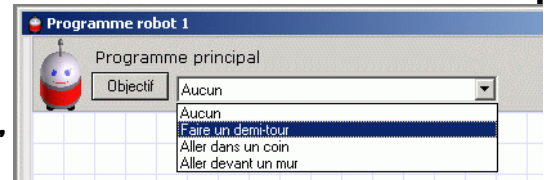
Remarques :

*L'objectif de l'exercice **Faire un demi-tour** est le suivant :*

A la fin de l'exécution, le robot doit occuper la même case qu'au début et doit être dirigé vers la direction opposée à celle du début.

Pour faire un demi-tour, il suffit de tourner deux fois de suite à droite ou à gauche.

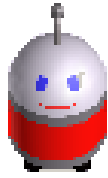
1- Dans la fenêtre programme, sélectionne **Faire un demi-tour** dans le menu déroulant.



2- Cliquez ensuite sur le bouton . Si l'organigramme est correctement réalisé, la mention **Organigramme correct** apparaîtra dans la fenêtre.

3- Dans la fenêtre **Exécution**, cliquez sur le bouton **Init** puis sur celui sur lequel figure le **petit robot**. Le programme s'exécute. Le robot tourne deux fois sur lui-même et saute de joie car il a réussi son demi-tour.

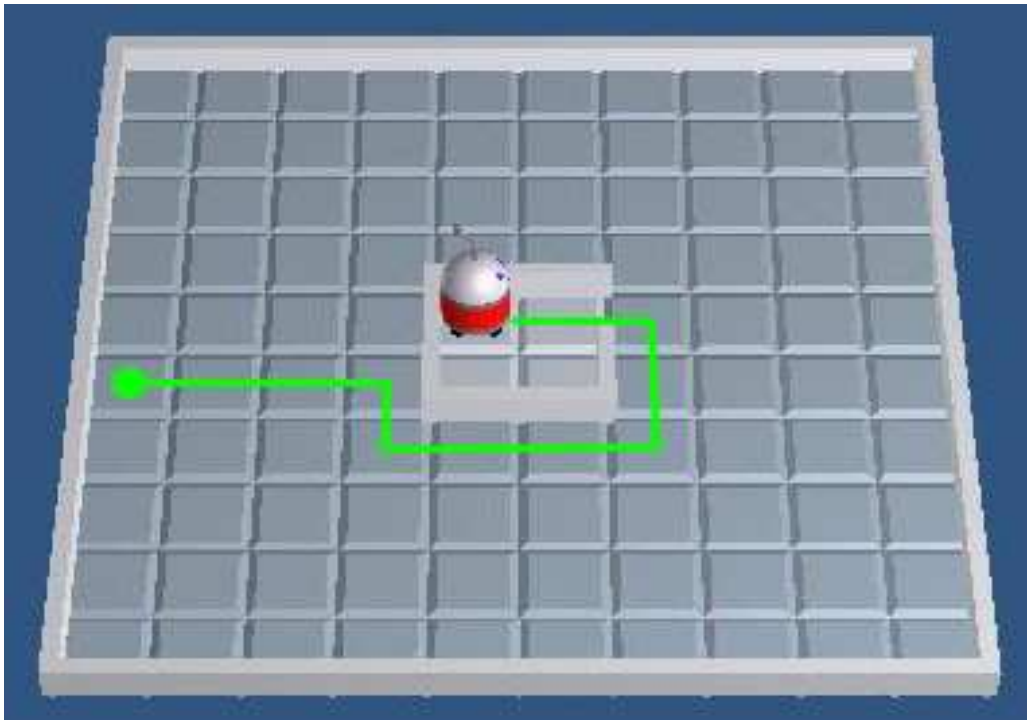
4- Pour faire réapparaître la fenêtre **Outils**, cliquez n'importe où dans la fenêtre **Programme**.



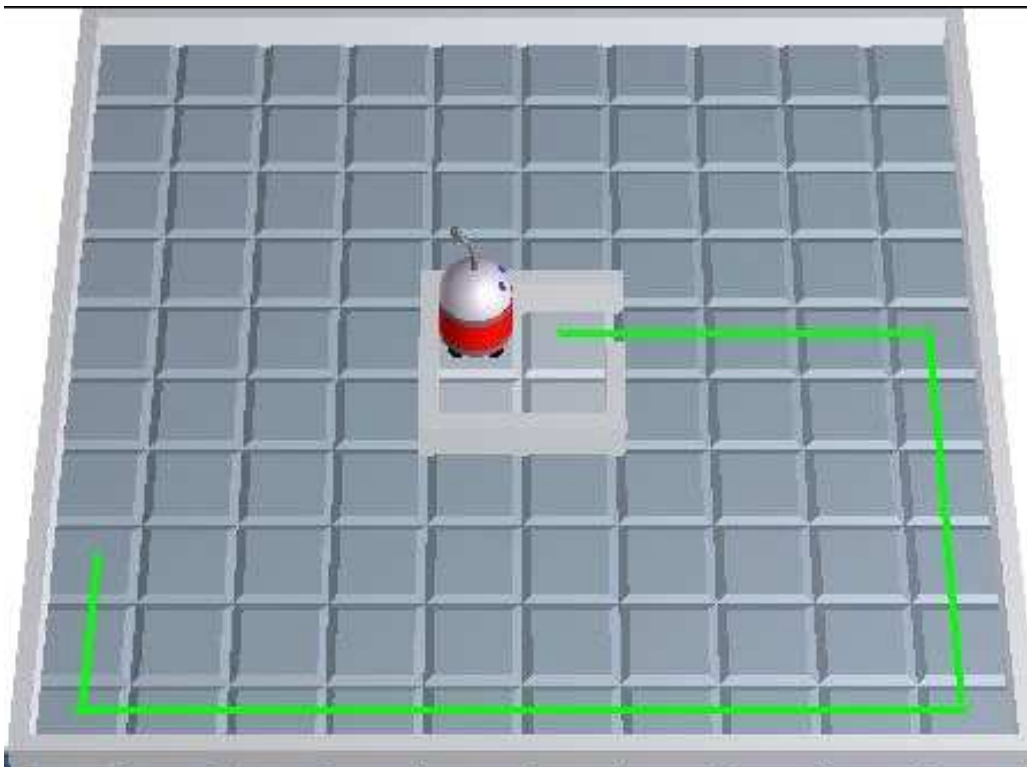
Faites parcourir à votre robot le trajet vert ci-dessous. Dans la fenêtre programme, vous pouvez sélectionner l'objectif Aller devant un mur. A la fin de l'exécution, RobotProg vérifiera si l'objectif a été atteint.

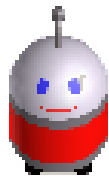
Sauvegardez ensuite votre programme.

Trajet n°1



Trajet n°2





Tests et conditions logiques

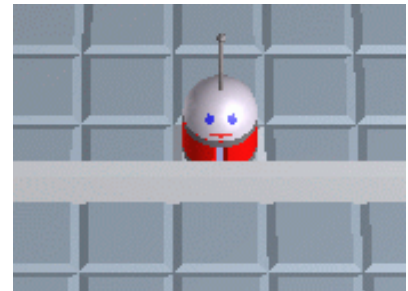
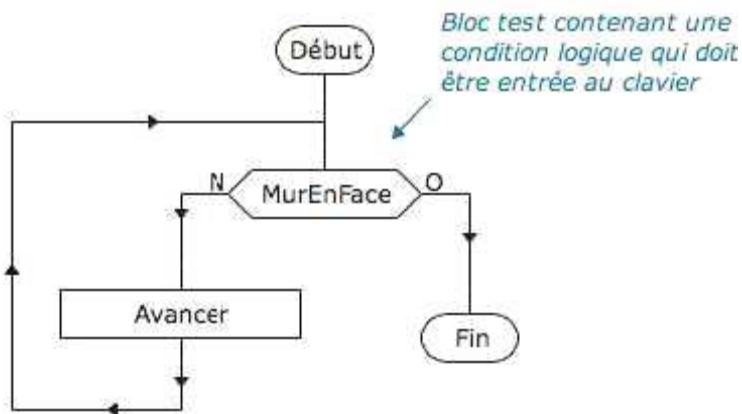
Fermez le programme précédent et choisissez le menu Fichier > Nouveau programme.

Ici vous allez construire un programme permettant d'aller jusqu'à un mur.

Choisissez l'objectif **Aller devant un mur**.

* **Programme contenant un test**

Construisez l'organigramme représenté ci-dessous et lancez son exécution.



Pour modifier le texte contenu dans un bloc test, il faut choisir l'outil sélection et faire un double clic sur le bloc.

La **condition logique** permet de vérifier une condition. Le résultat peut être vrai ou faux.

Si le résultat est vrai, l'exécution se poursuit après la sortie marquée O (oui ou vrai) ;

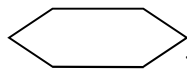
si le résultat est faux, l'exécution se poursuit après la sortie marquée N (non ou faux)

Dans cet exemple, la condition logique est **MurEnFace**, c'est un mot-clef du langage du robot qui fournit un résultat de type logique (vrai ou faux) en fonction de la position du robot au moment où elle est vérifiée

Si le robot est en face d'un mur, le programme s'arrêtera, sinon le robot avancera et recommencera le test.

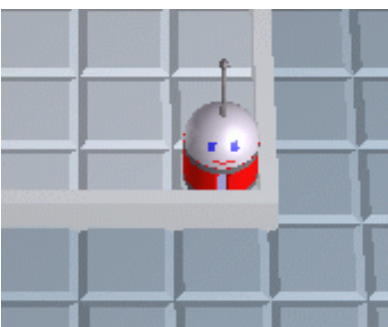
IV Exercices :

tu vas devoir utiliser le symbole .



Il peut contenir les questions suivantes :

MurADroite (Y a-t-il un mur à droite ?) ; MurAGauche (Y a-t-il un mur à gauche ?) MurEnFace (Y a-t-il un mur en face ?)



Aller dans un coin :

Objectif : A la fin de l'exécution, le robot doit se trouver dans un coin. Il peut avoir une direction quelconque. La case finale doit être différente de la case initiale.

Dans la fenêtre **Programme**, sélectionne **Aller dans un coin** dans le menu déroulant

La méthode pour aller dans un coin peut être la suivante : le robot va d'abord devant un mur,

puis il longe le mur, jusqu'à ce qu'il arrive en face d'un autre mur.

Pour tester si le robot vient d'arriver dans un coin, on peut utiliser la condition logique :

MurEnFace Et (MurADroite Ou MurAGauche)