

Activité : Sortir d'un labyrinthe en Scratch

B. Nguyen (INSA CVL)

Partie I : Algorithme de sortie du labyrinthe

Nous allons coder un algorithme basique (qui ne marche pas tout le temps) : suivre le mur droit d'un labyrinthe. Pour ce faire, nous allons programmer un lutin (lutin 3) qui devra disposer de la capacité de savoir si y a un mur à sa droite (fonction **tester si mur à droite**). Cette fonction met la valeur de la variable 'Mur à droite' à 1 (vrai) s'il y a un mur à droite et à 0 (faux) s'il n'y a aucun mur à droite.

Les fonctions données sont :

Se placer en position départ : se met au point de départ dans la bonne direction

Avancer : fait avancer le lutin

Reculer : fait reculer le lutin

1) Fonction mur à droite

Nous allons utiliser un test de Scratch qui permet de détecter si notre lutin touche un élément d'une certaine couleur :



Nous pouvons également modifier la valeur de la variable mur à droite (onglet données)



L'idée pour savoir s'il y a un mur à droite c'est de tourner à droite, avancer, et voir si on a touché un mur. Il faudra ensuite (selon qu'on aura touché un mur ou pas) mettre la bonne valeur dans la variable Mur à droite, puis se remettre dans la position précédente en reculant et en tournant à gauche.

2) Programme principal

Nous cherchons à rejoindre la sortie du labyrinthe (toucher la couleur rouge). Nous allons donc faire une boucle *répéter jusqu'à ...* de ce style :



L'idée de l'algorithme est la suivante :

- 1- On teste s'il y a un mur à droite
- 2- S'il n'y a pas de mur à droite on tourne et on avance
- 3- On avance
- 4- Si on a foncé dans un mur, alors on recule et on tourne à gauche

Faites tourner cet algorithme « à la main » pour voir s'il fonctionne bien.

Réalisez-le en Scratch.

Partie II : Algorithme de sortie de labyrinthe aléatoire

Programmez un algorithme qui avance jusqu'à ce qu'il tombe sur un mur, et dans ce cas, il tourne aléatoirement à gauche ou à droite.

Question subsidiaire / débat : est-ce qu'on est sûr avec un tel algorithme de pouvoir sortir d'un labyrinthe ?