Fiche 9 : Niveau 1-Etape3

Dans cette troisième étape, nous rajoutons la base, la dune, la végétation et le volcan. De plus, lorsque le rover arrive sur un bord, au lieu de rebondir, il doit réapparaitre de l'autre côté.

Préparation du projet

- Ouvrir le projet Session8.sb2 et le sauvegarder sous le nom « Session9.sb2 »
- Créer un lutin « dune », un lutin « végétation », un lutin « volcan » et un lutin « base ». Les placer n'importe où sur la scène.

Le rover ne rebondit plus sur les bords

Nous voulons que lorsque le rover touche un bord, il réapparaisse de l'autre côté.

Dans la catégorie mouvement, nous avons deux variables à disposition : « abscisse x » et « ordonnée y » qui contiennent la valeur des coordonnées x et y du lutin concerné.

Si abscisse x du rover > 240 **alors**

| donner la valeur -240 à x

fin si

Si abscisse x du rover< - 240 **alors**

| donner la valeur 240 à x

fin si

On réalise un algorithme identique pour le haut et le bas de la scène.

Rencontre avec la dune ou la végétation

Lorsque les lutins « dune » ou « végétation » sont touchés par le rover ils envoient un message « dune touchée » ou « végétation touchée ». Lorsque le rover reçoit ces messages il retourne à la base, ce qui lui fait perdre du temps.

Rencontre avec le volcan

Lorsque le lutin « volcan » touche le rover, il envoie un message « volcan touché », diminue le nombre de vie. Lorsque le rover reçoit ce message, il retourne à la base.

S'il n'a plus de vie, le volcan envoie le message « fin » afin que tous les lutins soient avertis de la fin du jeu, l'arrière-plan bascule sur « sol_martien3 » et tout s'arrête.

Empêcher l'eau et la nourriture de se positionner sur un des obstacles : volcan, dune, végétation.

Lorsque les ressources (eau, nourriture) sont récoltées, elles réapparaissent aléatoirement sur la scène. Il est parfaitement possible qu'une ressource réapparaisse à l'endroit où un piège (végétation, dune ou volcan) se trouve déjà, ce qu'il faut éviter car on ne peut pas avoir 2 objectifs contradictoires : récolter la ressource et éviter l'obstacle.

Comment faire en sorte que ce cas de figure ne puisse pas se produire ?

Il faut tirer au hasard une nouvelle position tant que la ressource touche un piège. L'algorithme devient :

1/ Placer la ressource sur un piège.

2/ Puis, effectuer la boucle suivante :

Répéter jusqu'à ce que la ressource ne soit plus sur un piège

| donner à la ressource une nouvelle position aléatoire

fin répéter