

Session 3 : Langage de programmation

Les machines qui nous entourent ne font qu'exécuter des ordres (instructions).

Pour donner des ordres à un ordinateur, on invente et on utilise des langages, que l'on appelle « Langages de programmation ».

Ces langages sont compréhensibles à la fois par la machine et par l'être humain.

Un langage de programmation est différent d'une langue naturelle :

- Il possède très peu de mots et de règles de grammaire
- Il ne laisse place à aucune ambiguïté
- Il nécessite une traduction pour que l'ordinateur puisse exécuter les instructions données dans le programme.

Il existe de nombreux langages de programmation, adaptés à différents usages.

Un bug est une erreur dans un programme.

Un bug minime en apparence peut avoir des conséquences énormes.

Inventez votre langage

- Distribuer la fiche³ et laisser les enfants inventer leur propre langage. Comparer les solutions proposées.
- Le langage que nous pouvons adopter :

Avance : avance d'une case en avant

Droite : pivote sur place d'un quart de tour vers la droite

Gauche : pivote sur place d'un quart de tour vers la gauche.

- Coder le parcours imposé avec ce langage : distribuer une grille de parcours, afin que les enfants puissent suivre au fur et à mesure les effets de l'instruction qu'ils écrivent.

Avec ce langage le programme du parcours imposé devient :

Avance ; Droite ; Avance ; Gauche ; Avance ; Gauche ; Avance ; Droite ;
Avance ; Droite ; Avance ; Avance ; Avance ; Droite ; Avance ; Gauche ;
Avance ; Droite ; Avance ; Avance ; Droite ; Avance ; Avance ; Droite ;
Avance.

Ne peut-on pas raccourcir le programme ?

Si l'on ajoute un paramètre, indiquant le nombre de case à traverser, à l'instruction Avance le programme devient :

Avance 1 ; Droite ; Avance 1 ; Gauche ; Avance 1 ; Gauche ; Avance 1 ;
Droite ; Avance 1 ; Droite ; Avance 3 ; Droite ; Avance 1 ; Gauche ;
Avance 1 ; Droite ; Avance 2 ; Droite ; Avance 2 ; Droite ; Avance 1.

Qu'est-ce qu'un bug

Si j'oublie une instruction, si j'intervertis deux instructions, si je me trompe d'instruction (je tourne à gauche au lieu de tourner à droite par

exemple), mon programme ne va pas faire ce que je veux qu'il fasse. Ici je vais sortir du parcours imposé et rater la base.

On appelle bug, une erreur dans un programme. La cause d'un bug peut être diverse :

- Erreur dans l'algorithme : la succession des instructions dans le programme n'est pas correcte, ou le programmeur n'a pas utilisé les bonnes instructions
- Erreur de syntaxe : erreur dans l'écriture d'une instruction qui empêche le programme de traduction de générer la version du programme compréhensible pour l'ordinateur ou qui provoque une mauvaise traduction.
- Erreur matérielle : panne d'un élément de l'ordinateur.

Avec Scratch

- Nous partons du projet dans lequel nous avons créé le lutin Rover et que les enfants ont sauvegardé lors de la session2.

Pour les enfants qui ne seraient arrivés à rien de bien, ou qui ont perdu leur projet, on peut exporter le rover d'un projet, et l'importer dans d'autres projets. On peut aussi importer un rover imposé à tous.

Si l'importation fonctionne mal, on peut aussi fournir aux enfants un projet dans lequel tout est déjà prêt.

- Si le rover est trop gros sur la scène, il est possible de le diminuer en utilisant le bouton « Réduire »




On peut aussi utiliser, en début de programme, la commande :



en utilisant une valeur inférieure à 100.

- Importer l'arrière-plan xy1_grid.png
- Expliquer aux enfants le système de coordonnées en x et y des pixels constituant l'arrière-plan.
- Commencer le script par le bloc chapeau « Quand le drapeau est

cliqué »  : expliquer aux enfants qu'il s'agit là, de la condition de démarrage du programme. Toutes les instructions constituant le programme, sont placées sous ce chapeau et s'exécuteront l'une après l'autre.

- Placer le rover en $x = -182$ et $y = -130$, bloc « aller en $x = y =$ ».



- Pour programmer le parcours rouge, le rover doit démarrer en regardant vers le haut. Bloc 

- Réaliser l'algorithme de déplacement du parcours à l'aide des blocs « avancer de » (on multiplie le nombre de pas par 100)



et « tourner à droite 90 degrés»



ou

« tourner à gauche 90 degrés»



Note : Un algorithme est une méthode permettant de résoudre un problème. Un programme est l'expression d'un algorithme dans un langage de programmation.

- Le programme s'exécute très vite et l'on n'a pas le temps de voir ce qui se passe. Intercaler entre chaque déplacement un bloc



- On ne voit pas le tracé du parcours. Il faut donc munir le rover d'un

crayon : bloc que l'on peut faire

précéder du bloc

- Sauvegarder le projet.