

TP 1

Table des matières:

- I. [Introduction](#)
- II. [premiers pas](#)
- III. [Tourner](#)
- IV. [Nouveau bloc](#)
- V. [Nouveau type de boucle](#)
- VI. [Les conditions](#)
- VII. [Evolutions des conditions](#)
- VIII. [Conclusion](#)

I. Introduction

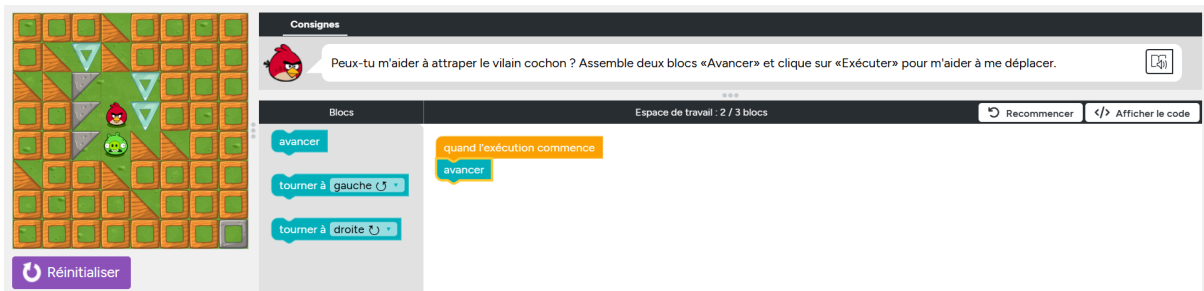
Ce travail pratique porte sur la programmation. Il va nous permettre de faire un premier pas en programmation en utilisant le langage JavaScript. Pour nous faciliter la tâche nous allons utiliser le site internet studio.code.org/hoc/1 qui nous permet d'utiliser des blocs comme scratch. Le défi du jour sera donc d'apprendre à réaliser toutes les missions proposées par ce site en emboîtant les blocs de façon à ce que l'algorithme fonctionne.

II. Premiers pas

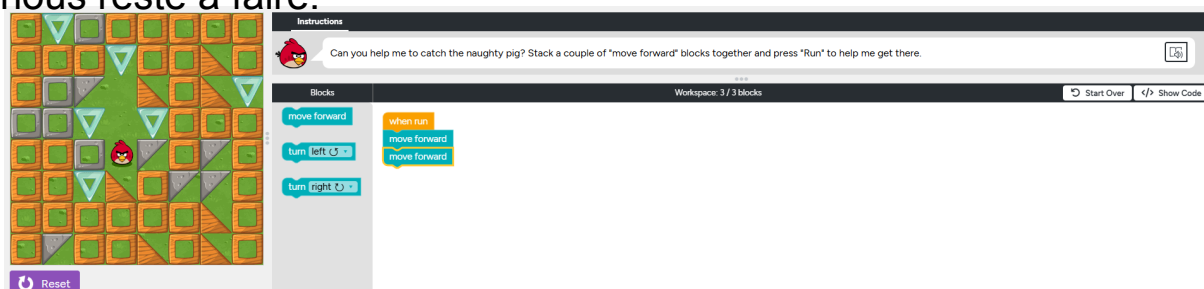
Nous commençons donc par le niveau 1 comme ceci:



Nous nous trouvons devant une interface très claire avec un objectif tout aussi clair : *nous devons déplacer l'oiseau jusqu'au cochon sans qu'il ne se retrouve bloqué par un mur*. Pour cela nous avons un 2 blocs déjà emboîtés le premier en j'aune qui permet de commencer le script et celui d'après qui permet de faire avancer l'oiseau. Egalement dans la colonne qui se trouve à gauche de la zone d'algorithme on peut y retrouver les blocs "tourner à gauche" et "tourner à droite" Lorsque je clique sur le bouton de lancement du programme il se passe ceci:

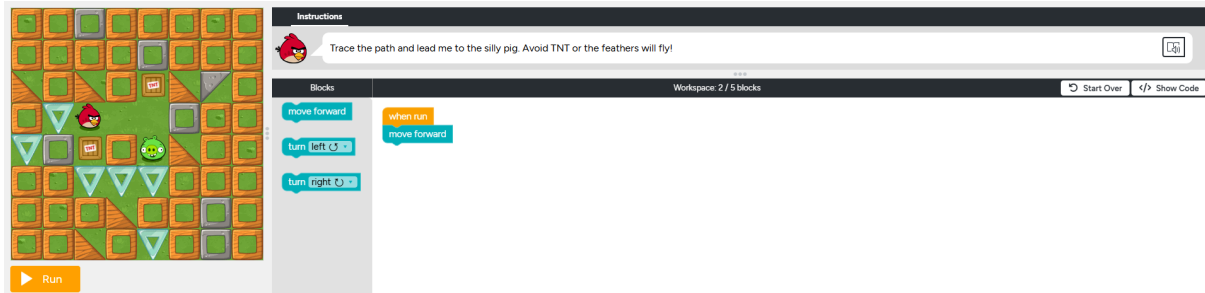


l'oiseau n'avance que d'une case on peut donc comprendre ce qu'il nous reste à faire:

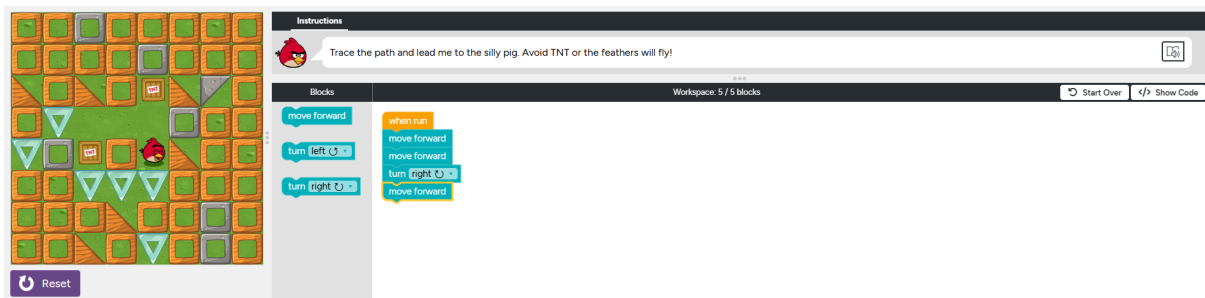


III. Tourner

Plus tard un labyrinthe à virage apparaît:



Le principe est plutôt simple il va falloir utiliser les blocs “tourner à gauche” ou “tourner à droite” et le placer au bon moment entre deux blocs “avancer” :

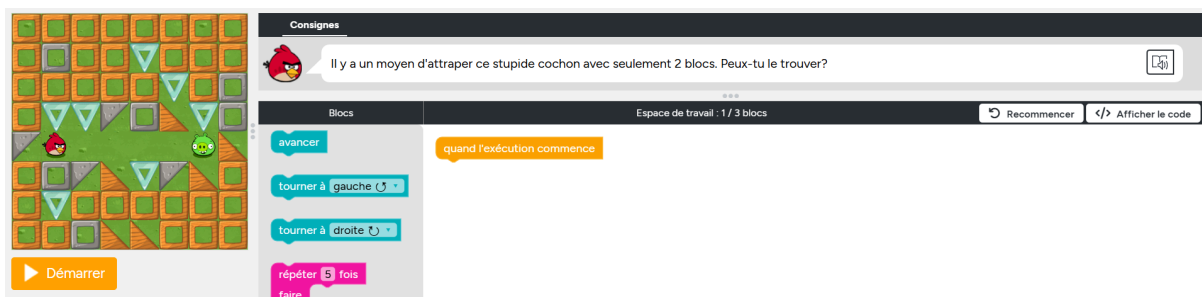


IV. Nouveau bloc

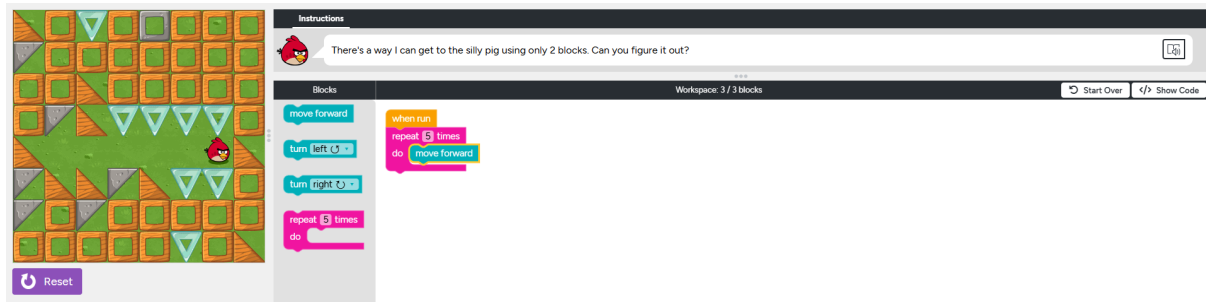
Au Niveau 6, Un nouveau type de bloc apparaît un bloc de type boucle numérique qui en javascript correspond à cette ligne:

```
for (var count = 0; count < 5; count++) {  
  }  
}
```

et le niveau ressemble à cela:



Ce bloc va nous permettre de réduire le nombre de bloc utilisé par exemple dans ce cas au lieu d'utiliser 6 blocs on nous allons en utiliser que 3:

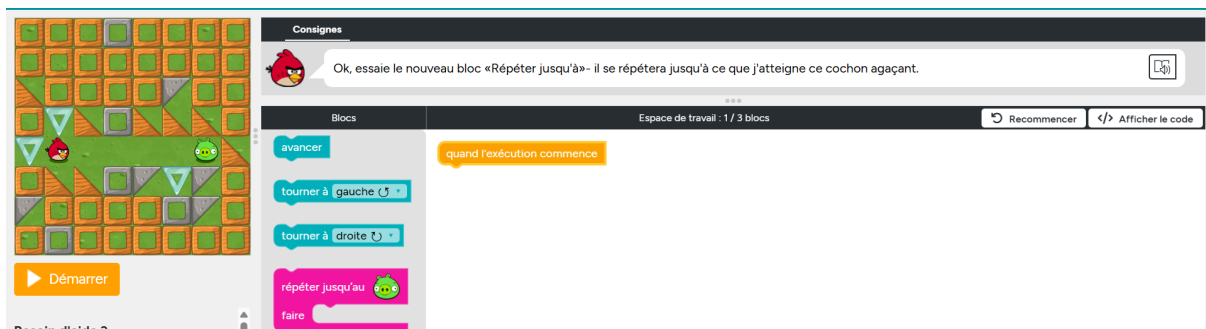


V. Nouveau type de boucle

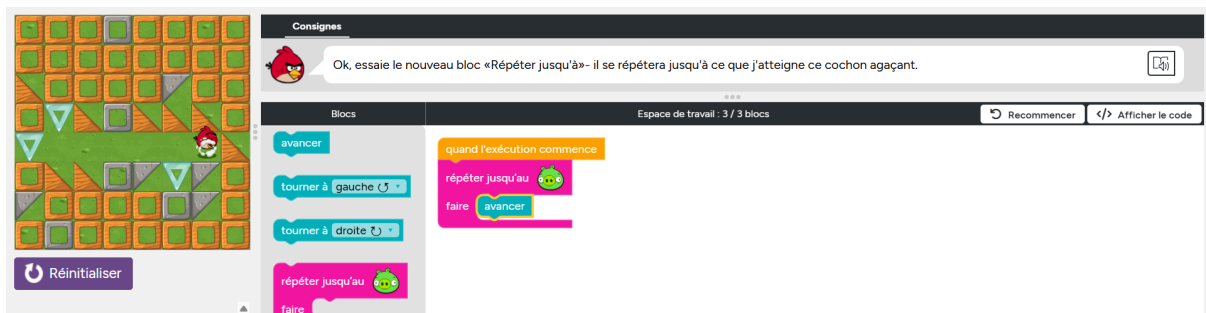
Au niveau 10 un nouveau type de boucle apparaît la boucle “jusqu’à” en Javascript elle correspond à ça:

```
while (notFinished()) {
}
```

et le niveau correspond à ça:

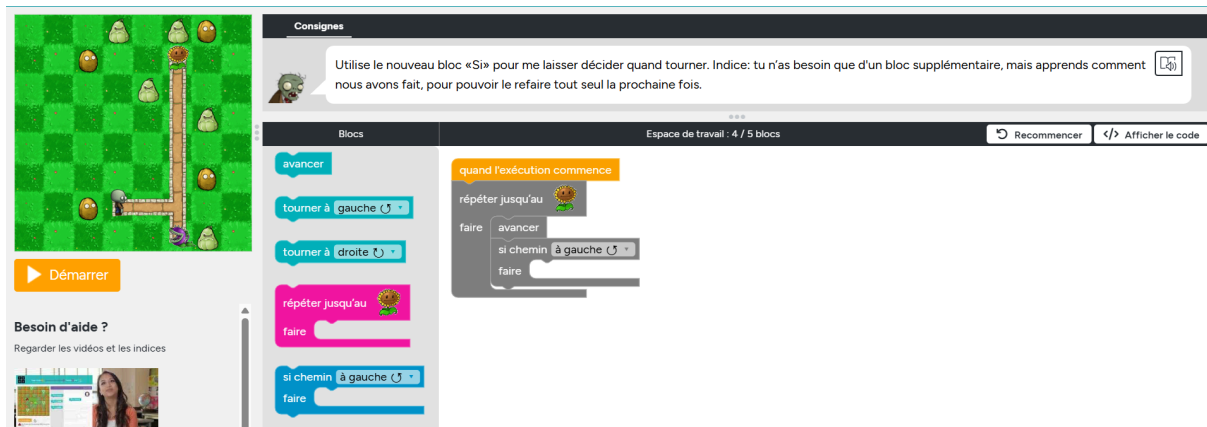


Le principe fonctionne de cette façon, jusqu'à ce que l'oiseau touche le cochon faire...
ce qui donne lorsque l'on résout l'exercice:

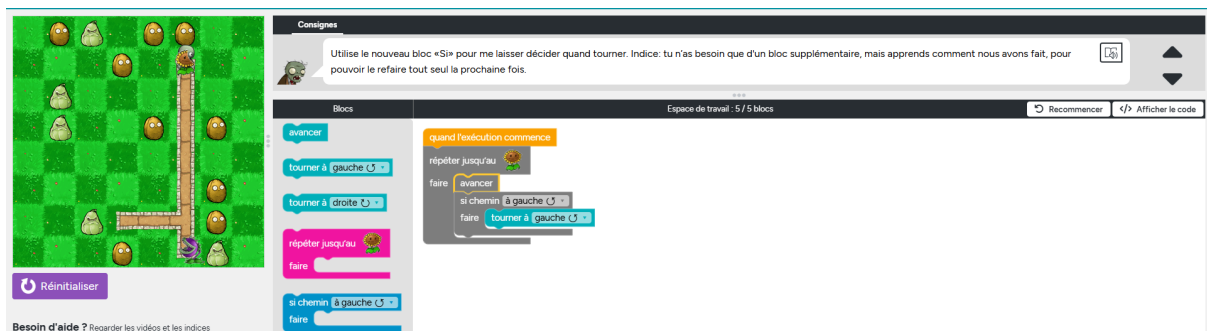


VI. Les conditions

Le nouveau bloc ajouté au niveau 14 est une condition:

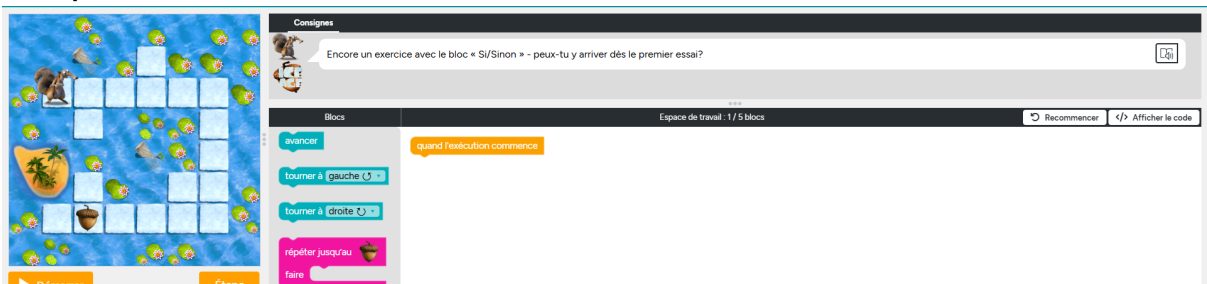


Ce bloc de condition permet des actions si la condition est remplie dans notre cas ce bloc détecte si un chemin est disponible et dans cet exercice je l'ai utilisé de cette manière:



VII. Evolution des conditions

Dans les deux derniers niveaux ces blocs là évoluent et laissent place à une autre condition dans le cas où la première ne serait pas remplie:



dans notre cas nous allons l'utiliser de cette manière:



VIII. Conclusion

Ce TP m'a permis d'apprendre des bases en JavaScript de manière très simple car l'emboîtement de bloc comme scratch se pratique depuis le collège et le fait de pouvoir regarder le code et donc d'apprendre également en langage direct en même temps. J'avais déjà des bases solides en programmation suite à ma spécialité NSI au lycée mais je n'ai jamais fait de JavaScript. Au fil des niveaux, j'ai appris à utiliser différentes structures fondamentales en algorithmique Java : les déplacements simples, les boucles, les conditions et leurs évolutions. Ces notions constituent les bases indispensables pour écrire des programmes plus complexes.