



# G0 - Epitech Diversity

B-G0-000

## Coding Club

GrinchKong



Bonjour à toi,

Aujourd'hui nous allons ensemble devoir sauver Noël ! Je sais que cela peut sembler bizarre mais récemment nous avons appris que le Grinch à enfermer le Père Noël dans un jeu vidéo reprenant bizarrement les trains de Donkey Kong ! Il faut vite faire en sorte que le jeu puisse de nouveau fonctionner et que tu parviennes à délivrer le Père Noël en complétant le niveau !

Prérequis :

<https://www.python.org/downloads/>

<https://www.pygame.org/wiki/GettingStarted>



## Le Python ?!

Je pense qu'avant de commencer il serait essentiel de voir un peu comment se présente le langage de programmation que tu vas utiliser, le Python. Pour cela on va faire un rapide tour global.

Commençons par les variables, en Python, les variables ne nécessitent pas de déclaration explicite pour réserver de l'espace mémoire. La déclaration se produit automatiquement lorsque vous assignez une valeur à une variable. Les variables peuvent être locales ou globales.

```
1 x = 5 # Variable globale
2 def some_function():
3     y = 10 # Variable locale
4     print(x, y)
5 some_function()
```

```
"Entiers :" num = 10
"Nombres à virgule flottante :" float_num = 10.5
"Chaînes de caractères :" text = "Bonjour"
"Booléens :" flag = True
```

Le Python supporte divers types de données, y compris les entiers, les nombres à virgule flottante, les chaînes de caractères (strings), et les booléens.

Pour ce qui est des listes, elles peuvent être redimensionnées dynamiquement et peuvent contenir des éléments de différents types. Le Python présente également des collections de paires clé-valeur, des collections immuables d'éléments qui peuvent être de différents types mais aussi des collections non ordonnées de valeurs uniques.

```
"Listes :" ma_liste = [1, "deux", 3.0]
"Dictionnaires :" mon_dico = {"cle1": "valeur1", "cle2": "valeur2"}
"Tuples :" mon_tuple = (1, "deux", 3.0)
"Ensembles :" mon_ensemble = {1, 2, 3}
```

```
1 for i in range(5):
2     print(i)
3
4 i = 0
5 while i < 5:
6     print(i)
7     i += 1
8
9 x = 10
10 if x > 5:
11     print("Plus grand que 5")
12 elif x == 5:
13     print("Égal à 5")
14 else:
15     print("Moins que 5")
```

Python prend en charge les boucles for et while, ainsi que les structures conditionnelles if, elif et else.



Les fonctions en Python sont définies en utilisant le mot-clé `def` et peuvent avoir des paramètres. Elles peuvent renvoyer des valeurs.

```
1 def ma_fonction(param1, param2):  
2     return param1 + param2  
3 resultat = ma_fonction(5, 3)
```

```
1 class MaClasse:  
2     def __init__(self, valeur):  
3         self.valeur = valeur  
4     def afficher_valeur(self):  
5         print(self.valeur)  
6  
7 mon_objet = MaClasse(10)  
8 mon_objet.afficher_valeur()
```

Python est un langage orienté objet, permettant de définir des classes et de créer des instances d'objets.

Python facilite la lecture et l'écriture de fichiers, avec un support intégré pour plusieurs formats de fichier.

```
1 with open("mon_fichier.txt", "r") as fichier:  
2     contenu = fichier.read()  
3     print(contenu)
```

Une grande force de Python réside dans sa vaste gamme de bibliothèques et de frameworks, qui s'étendent sur divers domaines comme l'analyse de données, le web, l'apprentissage automatique, etc.

```
1 import random  
2 print(random.randint(1, 10))
```

```
1 def ma_fonction():  
2     if True:  
3         print("Bonjour")
```

La syntaxe de Python est claire et concise, ce qui facilite l'apprentissage et la lecture du code.

Voilà je pense que tout est dit et qu'avec cela tu pourras mieux comprendre ce que tu vas faire, maintenant passons à ce qui nous intéresse le plus, faire en sorte que le Père Noël puisse enfin être délivré du Grinch !

Mais avant tu devrais aussi retrouver quelques fonctions utiles via ce lien :

[Pygame Documentation](#)



Cela te redirige vers la documentation de la librairie graphique Pygame, tu auras besoin de certaines de ces notions afin de pouvoir compléter le code qui t'es demandé !

Bien, après toutes ces longues explications, on peut enfin commencer ! Le Père Noël a besoin de nous et au plus vite, surtout que s'il reste menotté par le Grinch, il ne pourra jamais livrer tous les cadeaux aux enfants du monde entier !

Je vais commencer par t'expliquer comment tu lances le jeu et comment tout cela marche :

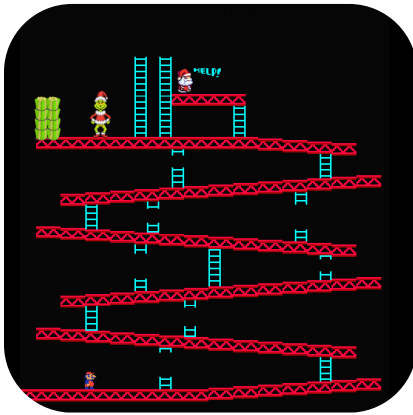
- Commence par regarder un peu à quoi ressemble le code pour t'y habituer un peu et le comprendre !
- Pour lancer le jeu, un cobra peut t'aider si jamais tu as du mal, mais tu peux soit appuyer sur la touche « F5 » si tu es sur « VSCode », soit l'exécuter via un terminal avec la commande python « grinch\_kong » ou « ./grinch\_kong » (seulement sur linux).

Bon si tout se passe bien et que tu as réussi, tu devrais atterrir sur une fenêtre où tu te retrouves sur le menu du jeu !

Bizarre, pourtant d'après les indications que j'ai si tu appuies sur « Espace » tu devrais te retrouver dans le jeu ?

Regarde de plus près le code mais je pense que quelque chose cloche dans la fonction « run », bizarre, tu peux y jeter un œil d'un peu plus près pour moi ?

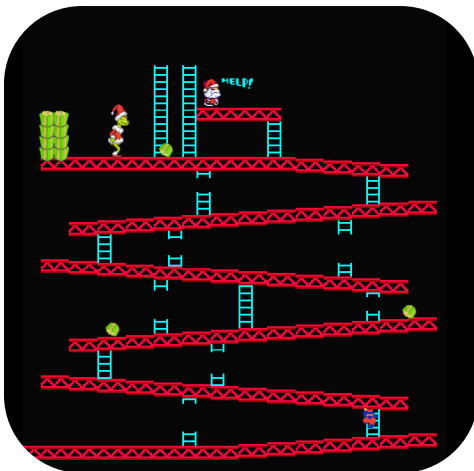
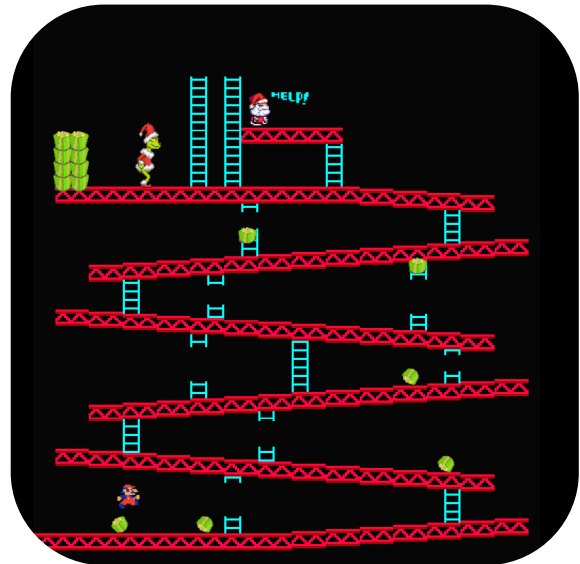




Ah tu as trouvé ! Tant mieux, voilà donc notre grand ennemi, le Grinch !

Aie, visiblement tu ne peux pas sauter, ni même monter les échelles, cela risque de poser un problème...

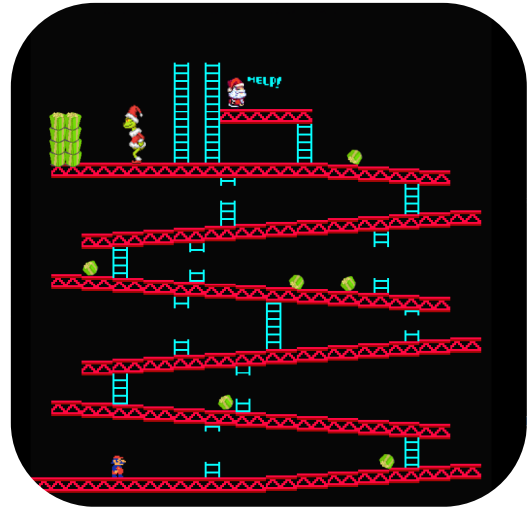
Bon essayons déjà de faire en sorte que tu puisses sauter ! Regarde la fonction « `jump_manager` », c'est celle qui s'occupe de la gestion du saut de ton personnage. Effectivement, elle ne contient rien qui puisse faire en sorte de créer un saut, hm, je pense que les commentaires peuvent t'aider à faire quelque chose, essaye de t'en servir au mieux !



C'est bon ? Ton personnage peut sauter ! Parfait, on va pouvoir faire en sorte que tu puisses maintenant de nouveau monter les échelles. Va dans la fonction « `ladderCheck` », elle sert à ce que ton personnage puisse se rendre compte de l'existence des différentes échelles ! Comme la fonction précédente, sert toi des commentaires pour t'aider à la compléter !



Mais c'est parfait ! Tu as réussi à faire en sorte que ton personnage puisse affronter le Grinch ! Seulement le chemin est ardu, le Grinch ne fais qu'envoyer que des cadeaux en boucle ! Faisons en sorte de réduire tout cela ! Je te laisse regarder la ligne 386 dans la fonction « game\_logic » et on va réduire les pouvoirs de ce méchant vilain !



Bon je pense que tout est bon tu dois être désormais capable d'arriver jusqu'au Père Noël ! Si c'est le cas bravo, tu as réussi à le délivrer !!! Mais bon faisons en sorte désormais que tout cela soit plus amusant !



Tu as désormais le choix de développer ce programme et de faire en sorte de l' étoffer ! Pour cela, tu peux rajouter toutes ces options :

- Un écran où il est indiqué que tu as gagné !
- Une gestion de score ! Et pourquoi pas l'affichage du meilleur score !
- Essayer d'implémenter un système de vie pour plus de défis
- Ou même encore rajouter des pièces !
- Bref tout ce qui te passe par la tête...



Si jamais tu veux faire autre chose ou même que tu as besoin d'aide pour cela, n'hésites pas à demander aux différents cobras leurs avis ou de l'aide !

