

OPENCLASSROOMS
DÉVELOPPEUR D'APPLICATION PYTHON :
PROJET 3 : AIDEZ MACGIVER À S'ÉCHAPPER!

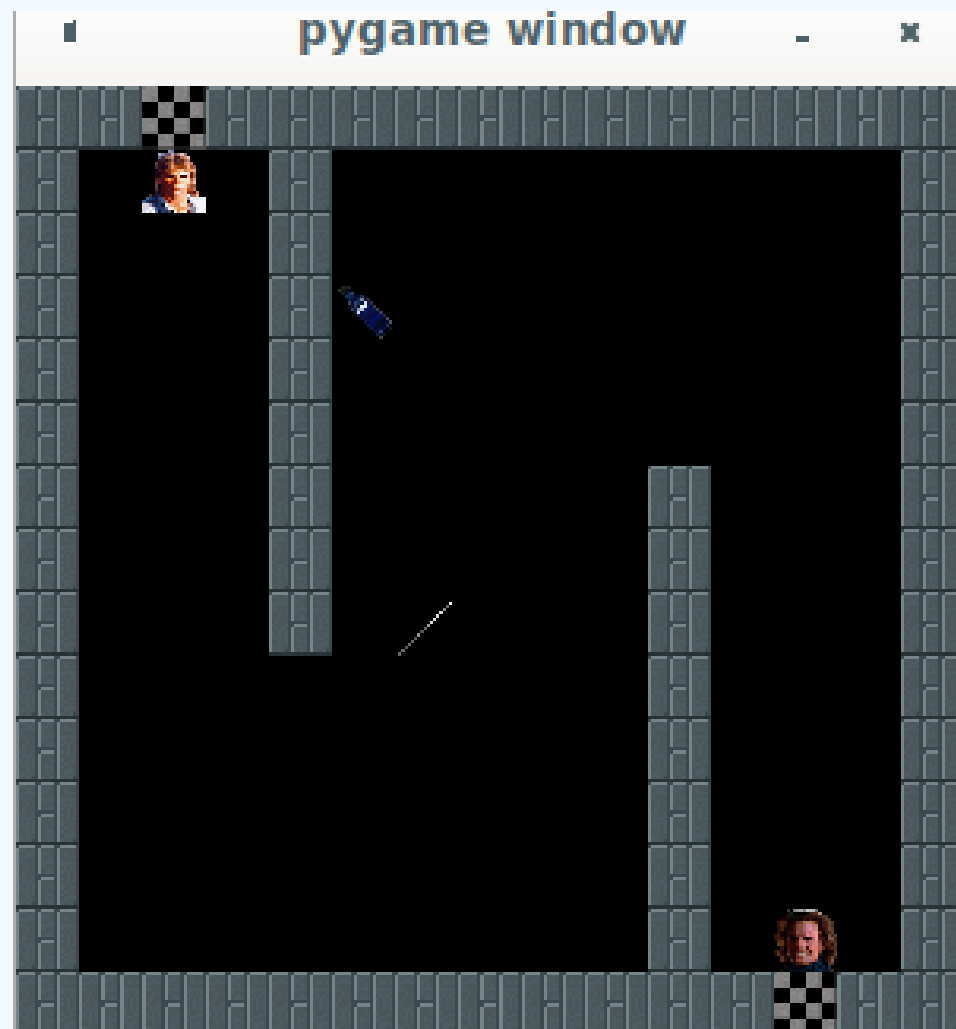
ÉVADEZ-VOUS DU LABYRINTHE!

Présenté par Céline PELLETIER

SOMMAIRE :

- Présentation du projet
- Outils
- L'algorithme:
 - Initialisation
 - Contrôle
 - Affichage du monde
- Les difficultés et solutions

PRÉSENTATION DU PROJET:



Projet de création d'un petit jeu vidéo en PYTHON.

Labyrinthe contenant 2 objets.

Le héro doit récupérer les 2 objets pour endormir le gardien et s'enfuir.

Si il n'a pas les 2 objets en arrivant devant le gardien :
il perd.

OUTILS :

PYCHARM

- IDE : langage PYTHON.
- permet le versionnage du code.
- permet de commit et push sur Github.
- permet de respecter la PEP 8 de manière intuitive.



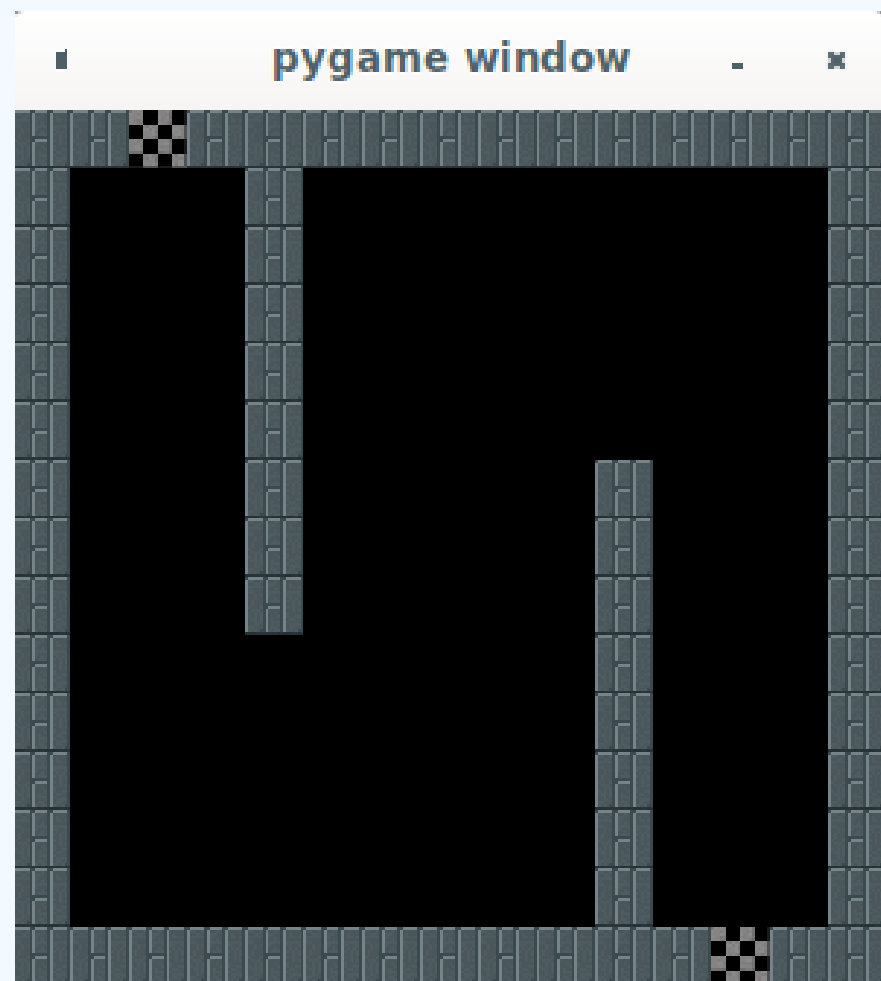
PYGAME

- Interface Graphique pour PYTHON.
- Permet l'affichage vidéo en 2D
- Permet la gestion de l'audio
- Permet de gérer les événements. (clavier, souris, collision etc...)



ALGORITHME

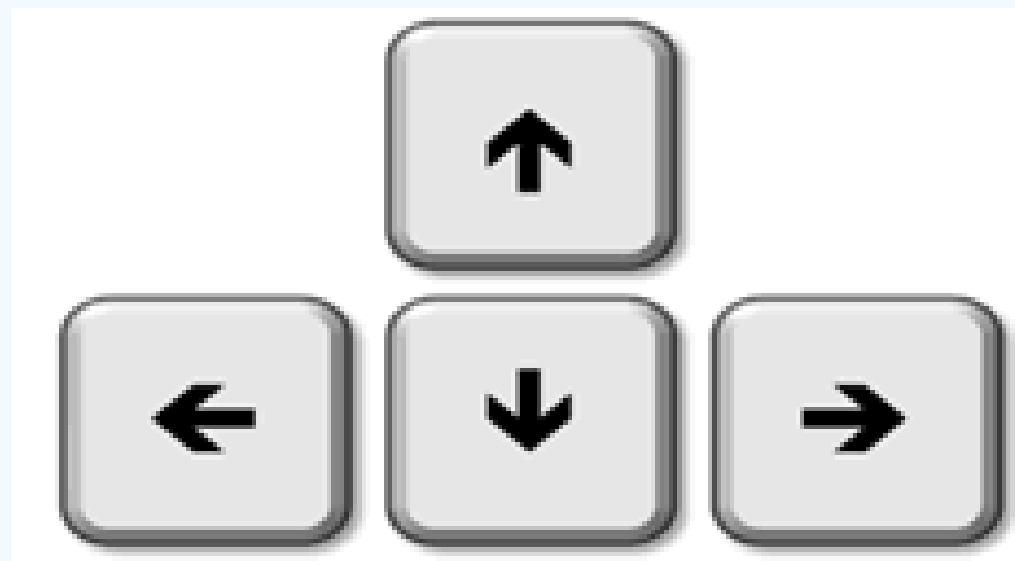
INITIALISATION :



- Création de l'écran.
- Création du labyrinthe.
- Génération de la configuration.
- Création des personnages et des objets.
- Suppression de la position de l'ether de la liste des positions vides.
- Remplacement du "P" en "x" dans la configuration.

ALGORITHME

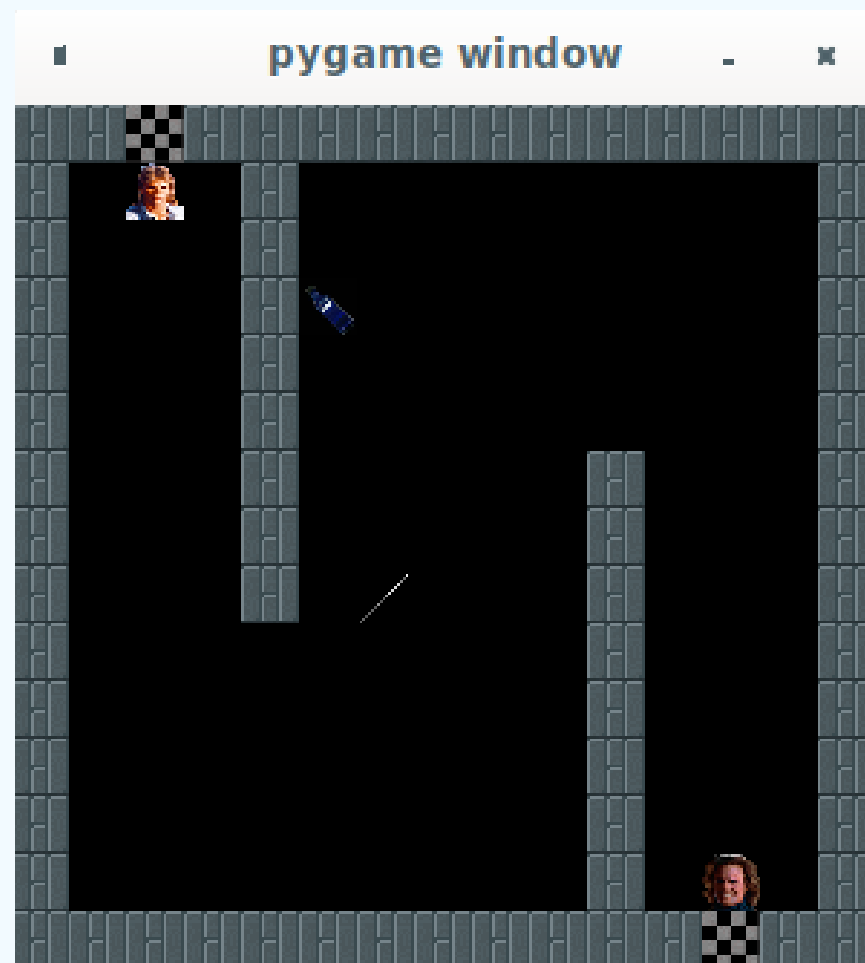
CONTRÔLE :



- Gestion des contrôles du héros.
- Gestion des collisions.
- Gestion de la récupération des objets du labyrinthe.

ALGORITHME

AFFICHAGE DU MONDE :



- Affichage du labyrinthe.
- Affichage des objets.
- Affichage du gardien.
- Affichage du personnage.

DIFFICULTÉS ET SOLUTIONS

LE LABYRINTHE

- Afficher des images.
- Afficher des sprites de la map de sols.
- Afficher plusieurs Sprites les uns à côté des autres.
- Créer un labyrinthe en créant des murs.



SOLUTIONS

- Créer un labyrinthe manuelle (lettre = Sprite), dans un fichier.
- Création classe "Lab".
- Fonction :
 - "generate_lab" : génère la configuration et 2 listes. (murs et positions vides)
 - "diplay_lab" : affiche le labyrinthe.

DIFFICULTÉS ET SOLUTIONS

LES OBJETS

- Afficher aléatoirement sans être sur les murs.
- Effacer quand le héros passe dessus.
- Image au bon format.



SOLUTIONS

- Création classe "Labobject".
- Création => redimension image.
- Fonction "draw_me": affiche.

DIFFICULTÉS ET SOLUTIONS

LE GARDIEN

- Affichage du gardien.



SOLUTIONS

- Création classe "Guardian".
- Fonction :
 - "draw_me" : affiche
 - "my_rect" : retourne le rectangle du gardien.

DIFFICULTÉS ET SOLUTIONS

LE HÉRO

- Mouvement du héros dans toutes les directions
- Affichage du héros.



SOLUTIONS

- Création classe "Player".
- Création => redimension image.
- Fonction :
"move_right", "move_left",
"move_up" et "move_down" :
déplace le héros.
- "draw_me" : affiche.
- Attribut obj1 et 2 == False.

DIFFICULTÉS ET SOLUTIONS

COLLISIONS

- Collisions avec les murs.
- Collisions avec les objets.
- Collisions avec le gardien.



SOLUTIONS

- Collisions murs :
liste des murs => collision =>
retour position précédente.
- Collisions objets :
ether, obj1 == True
needle, obj2 == True
- Collisions gardien :
si obj1, obj2 == True => Gagné.
sinon => Perdu.

DIFFICULTÉS ET SOLUTIONS

HÉRO ET GARDIEN

- Utilisation de leur lettre pour les créer.



SOLUTIONS

- Fonction "create_character" : crée le personnage à la position donnée dans la configuration du labyrinthe, et de la classe donnée.

DIFFICULTÉS ET SOLUTIONS

IMAGE DU HÉRO AU DÉPART

- Image du héros qui reste au départ.



SOLUTIONS

Fonction :
"erase_pos_character" :
remplace "P" par "x" dans
la configuration.

MERCI DE VOTRE
ATTENTION.

Céline PELLETIER

