

Descriptif du projet (Algorithmes et prog. Python)

1. Travail à réaliser

Un labyrinthe est un tracé sinueux, muni ou non d'embranchements, d'impasses et de fausses pistes, destiné à égarer ou à ralentir celui qui cherche à s'y déplacer. D'un point de vue abstrait, on appellera labyrinthe une grille à deux dimensions avec deux types de cases : les cases libres (blanches) et les murs (noirs).

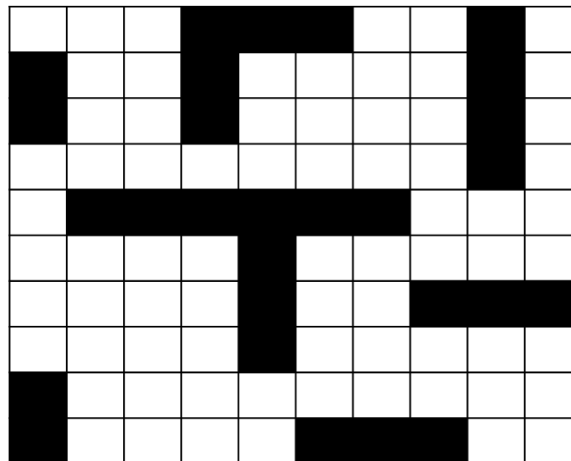


Figure 1: Exemple de labyrinthe.

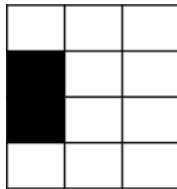
Partie 1 : (4 Points)

Le but de cette partie est d'implémenter une grille (labyrinthe) et les fonctions permettant de déterminer ou modifier les statuts des cases (blanches ou noires).

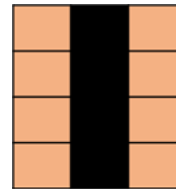
- 1) Comment modéliser informatiquement une telle grille ? Justifiez vos choix.
- 2) Donner l'implémentation correspondante. Il faut prendre en considération plusieurs points (les dimensions, la position de chaque case, le type de cases, ...). Le nombre de cases noires et leurs positions doivent être générés de manière aléatoire.
- 3) Afficher la grille en utilisant un module Python de votre choix : matplotlib, tkinter, seaborn,

Partie 2 : (4 Points)

Le but de cette partie est d'implémenter un algorithme permettant de déterminer si toutes les cases blanches d'un labyrinthe sont connectées entre elles. On considère que deux cases sont connectées si elles sont contigües horizontalement ou verticalement (pas en diagonale).



Les cases blanches sont connectées

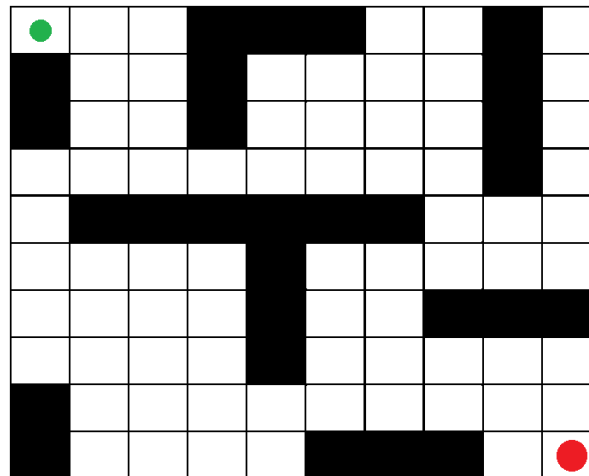


Les cases blanches ne sont pas connectées

- 1) Proposer un algorithme permettant de vérifier si toutes les cellules blanches sont connectées ?
- 2) Expliquez le fonctionnement de cet algorithme.
- 3) Donner l'implémentation correspondante.

Partie 3 : (8 Points)

Le but de cette partie est d'implémenter un algorithme permettant de trouver le plus court chemin pour se déplacer du point vert au point rouge.



- 1) Quelle structure de données est associée à l'ensemble des chemins traversés pour arriver à la destination ? Comment représenter informatiquement cette structure ?
- 2) Proposer un algorithme permettant de trouver tous les chemins pour atteindre la destination, puis indiquer le plus court chemin (Penser à utiliser les arbres binaires).
- 3) Expliquez le fonctionnement de cet algorithme.

- 4) Implémenter cet algorithme.
- 5) Adaptez la méthode d'affichage pour pouvoir visualiser les chemins dans la grille (En indiquant le chemin le plus court).

Partie 4 : (4 Points)

Le but de cette partie est de tester l'implémentation.

- 1) Au final, à partir des dimensions de la grille données par l'utilisateur, votre programme affichera la grille initiale et la grille finale (avec les chemins).
- 2) Enregistrer chaque exécution dans un fichier image contenant le nom d'utilisateur, la grille initiale et la grille finale.

Partie 5 : Bonus

Adaptez votre code pour qu'il trouve le chemin le plus court pour un point de départ et un point de destination générés de façon aléatoire.

2. Modalités du travail à effectuer

- Il s'agira de réaliser le travail demandé dans un fichier jupyter en commentant chaque partie de votre réalisation.
- Le travail et le rapport peuvent être réalisés seul ou en binôme.
- Le rapport sera un fichier .pdf avec comme nom : votre nom et votre prénom.
- Votre rapport doit être structuré et correctement présenté, incluant vos nom et prénom.
- **Votre travail doit être original !!**

Remarque : Vous pouvez échanger des idées entre binômes, voire travailler ensemble, mais le travail final présenté doit être spécifique et maîtrisé par les deux membres du binôme (vous devrez répondre à des questions sur le fonctionnement et le code).

2.1. Réalisation

Vous devez réaliser votre travail en utilisant la programmation orienté objet Python. Créez le code progressivement en le testant systématiquement à chaque étape. Vous pouvez par exemple commencer par créer une grille de cases vides, en testant toutes les fonctionnalités d'affichage et autre, puis ajouter les cases noires, en testant à nouveau, puis complexifier petit à petit. Conservez les versions qui fonctionnent bien même si elles sont en deçà de ce qui est demandé.

Si TOUT ce qui est demandé fonctionne, vous pouvez ajouter d'autres fonctionnalités et des options.

Afin de gérer les versions de votre programme et de permettre à votre enseignant de suivre le progrès de votre réalisation, il faut utiliser **github**. C'est l'opportunité de savoir comment utiliser cet outil pour organiser et gérer les projets. Le lien suivant vous présente les bases pour commencer avec github : <https://docs.github.com/fr/get-started/start-your-journey/hello-world>.

Suivez les étapes de ce lien : <https://docs.github.com/fr/repositories/managing-your-repositorys-settings-and-features/managing-repository-settings/managing-teams-and-people-with-access-to-your-repository> pour m'ajouter (nom d'utilisateur : mrkasri) comme contributeur à votre repo.

2.2. Démonstration

La démonstration se situe dans un créneau de 10 minutes comportant votre démonstration et les questions de l'enseignant. Elle est destinée à présenter les principales fonctionnalités de votre réalisation.

Il est important de faire attention aux points suivants :

- **Respectez** le temps imparti.
- Si vous avez réalisé votre travail sur votre ordinateur personnel, **testez-le** au préalable sur les ordinateurs de l'école ou apportez votre ordinateur.
- Ne vous perdez pas dans les détails techniques en parlant de votre travail. L'enseignant vous posera des questions techniques s'il le juge utile.
- Structurez votre démonstration autour d'un essai, c'est-à-dire des exemples qui montrent bien ce que fait votre programme.
- Tous les participants au projet doivent prendre la parole de manière à peu près égale.

3. Échéances et conseils

- Le code du projet et d'autres documents ou fichiers éventuels que vous souhaiteriez communiquer doivent être envoyés au plus tard le **01/04/2024 à 23h**.
- **Tout travail incomplet ou envoyé après la date limite ne sera pas accepté !!! Dans ce cas les étudiants seront déclarés absents.**
- Il est conseillé de ne dépasser les fonctionnalités de base que si ces dernières sont bien réalisées.
- L'ajoute d'autres fonctionnalités est fortement recommandé.