

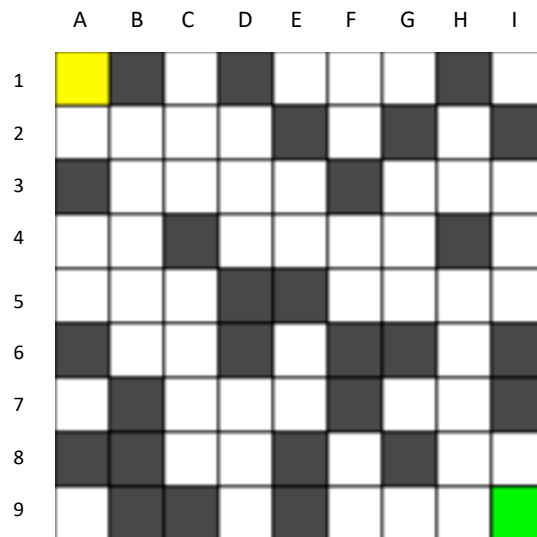
# Algorithmes sur les graphes – exercice

## PAR ICI LA SORTIE !

Les parcours de graphes (BFS et DFS) sont des algorithmes particulièrement adaptés pour la recherche du *plus court chemin* dans un graphe non pondéré et non orienté.

Nous allons l'illustrer par la mise en œuvre d'un algorithme de recherche du plus court chemin vers la **sortie d'un labyrinthe** en utilisant le parcours en profondeur (BFS).

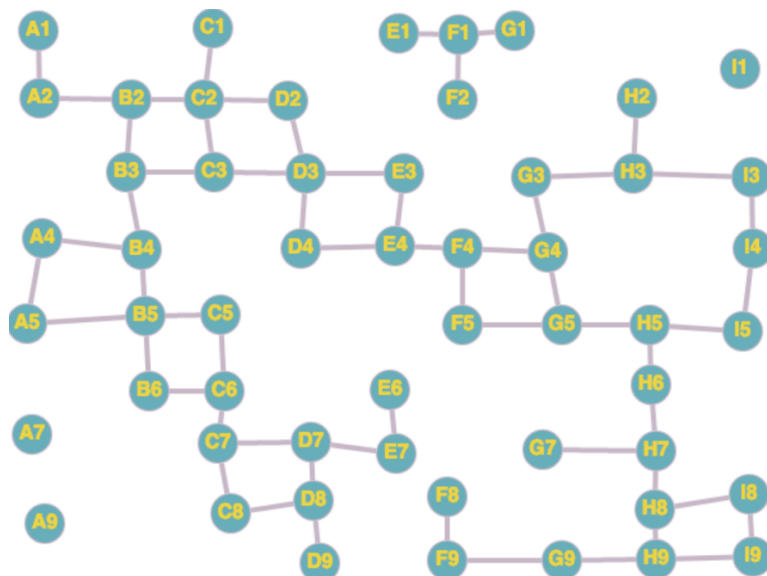
Voici le labyrinthe en question. Il s'agit de trouver le plus court chemin de la case A1 vers la case I9 en ne passant que par les cases blanches.



Retrouvez la version animée à l'adresse (maze3): <https://seanperfecto.github.io/BFS-DFS-Pathfinder/>

La partie code se trouve dans le fichier `labyrinthe.py`

Le labyrinthe peut se modéliser par un graphe dans lequel chaque sommet correspond à une case blanche et chaque arête à un chemin possible. Voici ce que cela donne :



Dans le fichier Python, on dispose du graphe modélisé sous forme de dictionnaire et de la fonction `bfs` de parcours en profondeur vue en cours.

## PREMIERE PARTIE : L'ARBRE COUVRANT DE POIDS MINIMAL

L'objectif est de modéliser l'*arbre couvrant* (représenté ci-dessous) sous la forme d'un dictionnaire généré par la fonction `bfs`.



Ce dictionnaire a pour clé un sommet et pour valeur la liste de ses voisins qui n'ont pas encore été visités.

❖ Modifiez la fonction `bfs` de manière à ce qu'elle retourne le dictionnaire modélisant l'arbre couvrant.

## DEUXIÈME PARTIE : LA RECHERCHE DU PLUS COURT CHEMIN

Maintenant que nous avons l'arbre couvrant, il suffit maintenant de partir de la sortie (I9) et de remonter de parent en parent jusqu'à revenir au sommet de départ.

❖ Modifiez la fonction `path` de manière à ce qu'elle retourne le sommet parent de la sortie I9 dans la liste `chemin`.

❖ Modifiez à nouveau la fonction `path` de manière à ce qu'elle retourne le sommet parent de la sortie I9 jusqu'à remonter de proche en proche au sommet de départ A1. On ajoutera les sommets dans la liste `chemin`.

Avez-vous trouvé le chemin le plus court ?