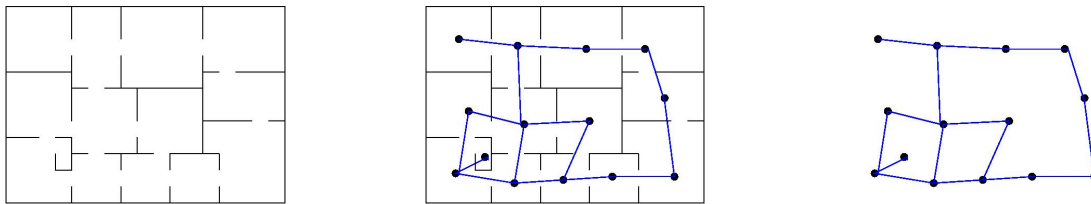


Objectifs :

- Créer un programme permettant de jouer au jeu des gardiens et voleurs.
- Se préparer au stage en effectuant un projet sur une courte période avec rapport et soutenance.

1 Règles du jeu

Ce jeu est un jeu à **deux joueurs** : les **gardiens** (joueur G) et les **voleurs** (joueur V). Le plateau de jeu est un musée que l'on représentera par un **graphe** : les sommets du graphe sont les salles et il y a une arête entre deux sommets si les deux salles correspondantes communiquent, comme dans l'exemple ci-dessous :



Ce jeu a plusieurs paramètres :

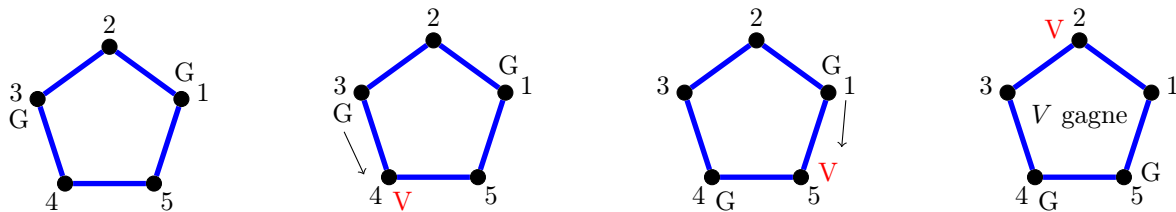
- Le musée-graphe : on peut jouer à ce jeu sur n'importe quel graphe. Pour commencer, on peut jouer sur un graphe simple comme un chemin ou un cycle.
- Le nombre de gardiens que l'on notera n_G .
- Le nombre de déplacements de gardiens autorisés par coup, noté n_D .
- Le nombre de voleurs qui peuvent arriver en même temps, noté n_V .
- Le nombre de coups maximum à jouer pour le voleur, noté n_C .

Pour les premiers jeux, nous conseillons de prendre : $n_D = n_V = 1$, n_C grand et n_G à trouver selon le musée !

Le jeu se déroule de la manière suivante :

1. Le joueur G place ses n_G gardiens dans les salles du musée.
2. Le joueur V positionne ses n_V voleurs sur des salles du musée.
3. Le joueur G peut faire jusqu'à n_D déplacements : un déplacement consiste à bouger un gardien d'une salle à une salle qui communique (en terme de graphes, d'un sommet à un sommet voisin).
4. Chaque voleur qui a un gardien dans sa salle est automatiquement arrêté et retiré du jeu. S'il reste des voleurs : **le joueur V a gagné**. Sinon, on recommence à l'étape 2 en laissant les gardiens dans leur nouvelle position : le joueur V replace des voleurs, puis le joueur G déplace ses gardiens...
5. Si au bout de n_C itérations, le joueur V n'a pas gagné, **le joueur G est déclaré gagnant**.

Exemple : Voici un exemple de déroulement du jeu : le musée a cinq salles disposées en cycle. Les valeurs des paramètres sont : $n_G = 2$, $n_V = 1$, $n_D = 1$, $n_C = 5$.



- 1) G place ses gardiens dans les salles 1 et 3.
- 2a) V place un voleur dans la salle 4
- 3a) G déplace son gardien en salle 3 vers la salle 4, le voleur est attrapé.
- 2b) On recommence : V place un voleur en salle 5.
- 3b) G décide de déplacer son gardien en salle 1 vers la salle 5, le voleur est attrapé.
- 2c) V place alors son voleur en salle 2 et gagne car aucun gardien ne peut l'attraper avec un seul déplacement.

2 Travail demandé

2.1 Interface graphique

Il vous est demandé de réaliser une interface graphique permettant de jouer à ce jeu à deux. Concernant les musées possibles, on pourra dans un premier temps avoir seulement quelques musées disponibles –en ligne, en grille, en cercle– dont les tailles peuvent varier.

Le programme doit reconnaître lorsqu'un joueur a gagné et proposer de faire varier les paramètres.

2.2 Jouer contre l'ordinateur

Nous vous demandons ensuite d'ajouter un mode pour jouer seul contre l'ordinateur. Il s'agit alors de créer une intelligence artificielle pour les gardiens et les voleurs. On pourra créer un programme plus malin pour les musées et paramètres simples, et un programme plus basique pour des paramètres plus compliqués.

2.3 Extensions possibles

Vous devez ensuite ajouter une ou plusieurs extensions à votre jeu :

- **Création de musée** : Permettre aux joueurs de créer son musée, les enregistrer. Développer une intelligence artificielle plus évoluée.
- **Plateforme multijoueurs** : Enregistrer les joueurs, organiser des tournois, se rappeler des scores...
- **Jeu évolutif** : Dans un soucis pédagogique, proposer plusieurs parties dont les paramètres sont fixés et dont le niveau varie. L'idée est d'amener l'utilisateur à découvrir des stratégies de jeu, à choisir le bon nombre de gardiens...
- **Jeu en réseau** (Mais je ne pourrais pas beaucoup vous aider)
- Ou tout autre idée que j'aurais au préalable validée.

3 Déroutement

Le projet s'effectue en **binôme** sur huit semaines (fin du projet avant la Toussaint). Attention, huit semaines c'est très court, ne perdez pas de temps !

Je serais présente 2 heures par semaine, mais 4 heures vous sont réservés et plus à la fin du projet. Profitez de toutes ces heures (et plus) pour avancer sur le projet.

Ne vous lancez pas tête baissée sur le projet, prenez le temps de bien lire le sujet, de réfléchir sur papier à la structure de votre programme.

3.1 Évaluation

Il y aura une évaluation au milieu du projet et une à la fin sous forme de mini-soutenance (10mn). Il faut rendre un rapport intermédiaire et un rapport final (environ 20 pages) qui expliquent votre démarche, les pistes et développements faits, la structure de votre programme. Vous devez aussi rendre l'ensemble de votre code sous la forme d'une archive (.zip de préférence) de sorte que n'importe qui, sur un ordinateur équipé d'une configuration similaire aux salles de TP, puisse exécuter votre code.

La note finale sera calculée ainsi :

- Évaluation mi-projet, coeff 1.
- Évaluation finale, coeff 2 : rapport 25%, soutenance 25%, réalisation 50 %

3.2 Dates

- Mardi 4 septembre : lancement du projet
- Mardi 2 octobre : évaluation mi-projet avec rapport intermédiaire en pdf. À cette date, il faut pouvoir jouer au jeu à deux et avec l'ordinateur, avec un minimum d'intelligence artificielle.
- Semaine du 22 octobre : fin du projet et évaluation.

En cas de questions/problèmes, n'hésitez pas à me contacter (aline.parreau@univ-lille1.fr).