

1. INTRODUCTION

Pour ce projet, vous devrez implémenter une version simplifiée du jeu vidéo "Blue Prince", en suivant les instructions données dans ce document. Une vidéo de démonstration du jeu final est fournie en plus de cet énoncé. Il est fortement recommandé de regarder la vidéo en plus de l'énoncé, il est difficile de comprendre le fonctionnement du jeu à partir de l'énoncé seul.

Les objectifs du projet sont:

- Mettre en pratique les concepts de POO vus en cours
- Gagner de l'expérience en programmation en Python
- Gagner de l'expérience en conception d'algorithmes
- Apprendre à collaborer sur des projets de programmation

L'évaluation du projet se fera à partir de trois éléments:

- Un court rapport présentant et justifiant vos choix de conception
- Une présentation orale suivie de questions
- Votre dépôt Github

Planning:

- 05/10: Annonce des groupes
- 14/11: Rendu du projet (rapport et dépôt Github)
- Semaine du 17/11: Examen oral

Questions: Si le présent document n'est pas assez précis pour vous permettre de comprendre le travail qui vous est demandé:

- Assurez vous d'avoir regardé la vidéo d'exemple donnée en plus de ce document
- Essayez de regarder sur internet (vidéos de streamers ou streameuses jouant au jeu, wiki du jeu https://blue-prince.fandom.com/wiki/Room_Directory)
- Enfin si vous avez toujours des questions, posez les moi par email

Plagiat: J'aurais accès à tous les rendus, et il sera donc facile de détecter les situations de plagiat, ou de collaboration étroite entre différents groupes. Ce serait évidemment lourdement sanctionné.

2. FONCTIONNEMENT DU JEU

Le jeu que vous devez implémenter est un jeu qui se joue en solo. Le joueur ou la joueuse se situe dans un manoir comportant 45 pièces disposées dans une grille de dimensions (5×9) . Ce manoir est "magique", et les pièces qui le composent sont choisies par le joueur ou la joueuse au fur et à mesure du jeu.

Quand le joueur ou la joueuse ouvre une porte menant vers une nouvelle pièce, le jeu propose de choisir parmi trois salles tirée au sort celle qui remplira cette position dans la grille. Le choix de la salle est très important pour progresser dans le jeu: elle peut comporter des portes offrant de nouveaux chemins, contenir différents objets, ou avoir un effet spécial. Le but du jeu est de progressivement construire un chemin menant vers la dernière pièce du manoir, située tout en haut de la grille.

Il existe deux manières de perdre:

- Le joueur ou la joueuse ne peut plus progresser dans le jeu: plus aucune porte ne peut mener vers l'arrivée, ou alors toutes ces portes nécessitent une clé et le joueur ou la joueuse n'en a pas.
- Le joueur ou la joueuse a épuisé son compte de pas. Les pas font parties des objets dans l'inventaire du joueur ou de la joueuse, ce qui est expliqué dans la prochaine section.

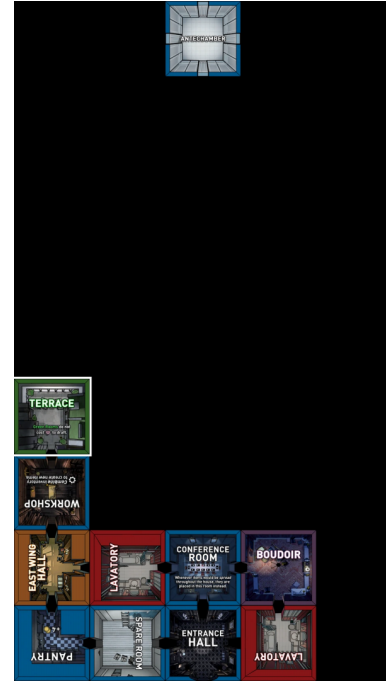


FIGURE 1. Grille de jeu comportant les différentes pièces du manoir. Les pièces en noir n'ont pas encore été choisies par le joueur ou la joueuse.

2.1. **Inventaire.** L'inventaire du joueur ou de la joueuse est composé de:

- Objets consommables:
 - Des **pas**: initialement à 70, le joueur ou la joueuse perd 1 pas à chaque déplacement (passer d'une pièce à une autre).
 - Des **pièces** d'or: initialement à 0, le joueur ou la joueuse peut ramasser des pièces dans le manoir, et les dépenser dans certaines salles en échange d'autres objets.
 - Des **gemmes**: initialement à 2, le joueur ou la joueuse peut ramasser des gemmes dans le manoir, et les dépenser pour choisir certaines salles lors du tirage au sort.
 - Des **clés**: initialement à 0, le joueur ou la joueuse peut ramasser des clés dans le manoir, et les dépenser pour ouvrir des portes fermées à clé, ou des coffres pouvant contenir des objets.
 - Des **dés**: initialement à 0, le joueur ou la joueuse peut ramasser des dés dans le manoir, et les dépenser pour tirer à nouveau au sort les pièces proposées lorsqu'on ouvre une nouvelle porte.

- Objets permanents: ces objets sont plus rares et fournissent un avantage permanent au joueur ou à la joueuse qui les trouve.
 - La **pelle**: permet de creuser à certains endroits, permettant de trouver certains objets.
 - Le **marteau**: permet de briser les cadenas des coffres, permettant de les ouvrir sans dépenser de clé.
 - Le **kit de crochetage**: permet d'ouvrir certaines portes, sans dépenser de clé.
 - Le **détecteur de métaux**: augmente la chance de trouver des clés et des pièces dans le manoir
 - La **patte de lapin**: augmente la chance de trouver des objets (y compris des objets permanents) dans le manoir
- Il existe d'autres objets dans le vrai jeu "Blue Prince" que vous pouvez également implémenter dans votre jeu si vous préférez. Vous pouvez aussi tout à fait implémenter d'autres objets issus de votre imagination, avec des effets de votre choix.






Inventory:	24 
Lockpick Kit	4 
Lucky Rabbit's Foot	2 
Shovel	2 
	0 

FIGURE 2. Capture d'écran d'un inventaire. A gauche, les objets permanents (kit de crochetage, patte de lapin, pelle). A droite, les objets consommables (pas, pièces, gemmes, clés et dés).

2.2. **Autres objets.** En plus des deux types d'objets précédents pouvant être ramassés par le joueur ou la joueuse, il y a dans le jeu plusieurs autres objets:

- La **pomme**: redonne 2 pas.
- La **banane**: redonne 3 pas.
- Le **gâteau**: redonne 10 pas.
- Le **sandwich**: redonne 15 pas.
- Le **repas**: redonne 25 pas.
- Les **coffres**: peuvent être ouverts avec une clé ou un marteau, et contiennent différents objets consommables.
- Les **endroits où creuser**: peuvent être creusés avec la pelle, et contiennent différents objets consommables (et parfois rien).
- Les **casiers**: uniquement dans le vestiaire (locker room en anglais), peuvent être ouverts uniquement avec des clés, et contiennent différents objets consommables (et parfois rien).

2.3. **Catalogue de pièces.** Le jeu dispose d'une liste de pièces, ayant des propriétés différentes, qui sont tirées au hasard à chaque ouverture de porte et proposées au joueur.

Chaque pièce est caractérisée par:

- Un nom (par exemple, "Entrance Hall", "Antechamber", "Vault", "Cellar", etc.)
- Une image pour la représenter dans la grille. Dans mon cas, j'ai directement récupéré les images disponibles sur le wiki du jeu: https://blue-prince.fandom.com/wiki/Room_Directory.
- Des portes menant dans différentes directions. Une pièce peut ne comporter qu'une seule porte (celle par laquelle elle a été découverte) et donc être un cul-de-sac.
- Un coût en gemmes que le joueur ou la joueuse devra dépenser pour pouvoir tirer cette pièce. Certaines pièces n'ont aucun coût.
- Des objets avec lesquels il sera possible d'interagir (ramasser, ouvrir, manger, creuser, etc). Certaines pièces ne contiennent aucun objet.
- Un effet spécial. Beaucoup de pièces n'ont aucun effet spécial. Des exemples d'effets sont donnés dans la section suivante.
- Un degré de rareté qui influence la probabilité de tirer la pièce. Ce degré est un entier entre 0 et 3. Chaque incrément du niveau de rareté doit diviser par trois la probabilité de tirer la pièce.
- Une condition de placement. Certaines pièces ne peuvent être tirées qu'à certains endroits. Par exemple, certaines pièces peuvent uniquement apparaître sur les extrémités du manoir.

Quand une pièce est ajoutée au manoir, elle est retirée de la "pioche" et ne peut plus être tirée. En revanche, vous pouvez choisir de mettre plusieurs exemplaires de la même pièce dans la pioche initiale du joueur. Dans l'exemple représenté dans la figure 1, on voit que la pièce "Lavatory" apparaît deux fois, elle était en effet en plusieurs exemplaires dans la pioche initiale.

2.4. Effets et couleur des pièces. Certains pièces possèdent des effets, et ces effets interagissent souvent avec les autres pièces du manoir en fonction de leur couleur. Dans le jeu, il existe plusieurs couleur de pièces:

- Les pièces **jaunes**: ce sont des magasins dans lesquels il est possible d'échanger de l'or contre d'autres objets.
- Les pièces **vertes**: ce sont des jardins d'intérieur, qui contiennent souvent des gemmes, des endroits où creuser, et des objets permanents.
- Les pièces **violettes**: ce sont des chambres, qui ont souvent des effets permettant de regagner des pas.
- Les pièces **oranges**: ce sont des couloirs, qui ont souvent beaucoup de portes.
- Les pièces **rouges**: ce sont des pièces qui ont souvent des caractéristiques ou des effets les rendant indésirables (peu de portes, retirent des pas, etc.).
- Les pièces **bleues**: ce sont les pièces les plus communes, avec des effets variés.

Les effets associés à chaque pièce sont indiqués directement sur l'image correspondant dans le wiki du jeu Blue Prince: https://blue-prince.fandom.com/wiki/Room_Directory.

Vous pouvez implémenter ces effets, ou les modifier en vous en inspirant (c'est ce que j'ai fait pour le jeu de démonstration), ou même proposer de nouveaux effets.

2.5. Déplacement. Le joueur ou la joueuse commence dans l'entrée du manoir (Entrance Hall) et doit rejoindre une pièce à l'opposé du manoir appelée l'antichambre (Antechamber).



(A) La pièce "Vault" contient 40 pièces d'or, a un degré de rareté de 3/3, coûte 3 gemmes, est bleue, ne comporte qu'une seule porte, et n'a pas d'effet spécial. <https://blue-prince.fandom.com/wiki/Vault>



(B) La pièce "Veranda" ne contient rien par défaut (mais il y a une probabilité élevée d'y trouver une gemme, un trou à creuser ou un objet permanent), a un degré de rareté de 2/3, coûte 2 gemmes, est verte, comporte deux portes, et a pour effet d'augmenter la probabilité de tirer d'autres pièces vertes. De plus, elle ne peut être placée qu'en bordure du manoir. <https://blue-prince.fandom.com/wiki/Veranda>

FIGURE 3. Deux exemples de pièces avec des caractéristiques différentes.

Un curseur représente la pièce actuelle du joueur ou de la joueuse dans le manoir. En utilisant les directions (ZQSD) le joueur ou la joueuse peut sélectionner une direction dans laquelle se diriger, et appuyer sur "espace" pour valider son choix:

- Si la direction donne sur un mur, alors rien ne se passe, le déplacement est impossible.
- Si la direction donne sur une porte déjà ouverte avec une pièce déjà découverte de l'autre côté, le joueur ou la joueuse se déplace dans cette nouvelle pièce.
- Si la direction donne sur une porte qui n'a pas encore été ouverte, alors le joueur ou la joueuse essaye d'ouvrir la porte. Si la porte est verrouillée à clé, alors le joueur ou la joueuse doit dépenser une clé pour pouvoir ouvrir la porte.

2.6. Ouverture d'une porte. Les portes peuvent avoir trois statuts: déverrouillée (niveau 0), verrouillée (niveau 1) et verrouillée à double tour (niveau 2). Une porte déverrouillée ne coûte pas de clé pour l'ouvrir. Les portes verrouillées et verrouillées à double tour coûtent une clé. Si le joueur ou la joueuse dispose d'un kit de crochetage, il ou elle peut ouvrir les portes de niveau 1 sans dépenser de clé. En revanche, le kit de crochetage ne permet pas d'ouvrir les portes verrouillées à double tour (niveau 2).

2.7. Tirage d'une pièce. Lorsque le joueur ou la joueuse ouvre une porte, le jeu tire trois pièces dans la pioche:

- Les pièces tirées au sort doivent pouvoir être tirées dans cette position (porte connectée à la pièce actuelle, pas de porte menant vers l'extérieur du manoir, autres conditions spécifiques de la pièce).

- Leur probabilité de tirage dépend de leur degré de rareté.
- Pour empêcher que le jeu soit complètement bloqué si le joueur ou la joueuse n'a pas de gemmes, il faut également s'assurer qu'au moins une des trois pièces proposées a un coût en gemmes égal à 0.

Si le joueur ou la joueuse possède des dés, et que le tirage n'est pas satisfaisant, il est possible de dépenser un dé pour retirer trois pièces dans la pioche.

Le joueur ou la joueuse choisit parmi les trois pièces à l'aide des touches directionnelles du clavier, et valide son choix avec "Entrée". Si la pièce choisie possède un coût en gemmes, ce coût est directement soustrait au compteur de gemmes du joueur ou de la joueuse. La pièce choisie s'ajoute à la grille du manoir, et il est maintenant possible d'y entrer.

2.8. Implémentation de l'aléatoire. Votre programme doit proposer une expérience de jeu nouvelle à chaque fois qu'on lance le programme. Vous devrez utiliser le module random pour vous assurer que plusieurs mécanismes sont aléatoires dans le jeu:

- Le statut d'une porte (niveau 0, niveau 1 ou niveau 2) est aléatoire, et la probabilité de trouver des portes avec des niveaux de verrouillage élevés augmente plus on avance dans le manoir. Sur la première rangée de pièces du manoir (la rangée où on commence), on ne doit tomber que sur des portes déverrouillées (niveau 0). Sur la dernière rangée de pièces du manoir, où se situe l'antichambre, on ne doit tomber que sur des portes verrouillées à double tour (niveau 2).
- Lorsqu'on ouvre une porte, les pièces proposées doivent être tirées aléatoirement.
- Certaines pièces peuvent comporter des objets avec une probabilité associée. Par exemple, le foyer (Den) comporte toujours une gemme, mais peut aussi parfois comporter un coffre.
- Certains effets de pièce (Veranda, Maid's Chamber), ou certains objets permanents (patte de lapin, détecteur de métaux), peuvent modifier les probabilités de trouver des objets.
- Certains effets de pièce (Furnace, Greenhouse, Solarium) peuvent modifier les probabilités de tirer certaines pièces.
- Enfin, lorsqu'on ouvre un coffre, un casier, ou qu'on creuse un trou, les objets obtenus doivent être tirés aléatoirement parmi différents lots possibles (que vous définirez comme vous le voulez).

3. INTERFACE GRAPHIQUE

L'implémentation de l'interface graphique utilise pygame. Vous pouvez importer ce package au début de vos fichiers, et l'utiliser pour dessiner des objets et récupérer les commandes via le clavier. Vous pouvez parcourir la documentation de pygame pour trouver des méthodes qui répondent le mieux à vos besoins.

Vous pouvez demander à un LLM de vous aider à construire une première version du programme en utilisant pygame pour construire la grille de jeu et des menus pour interagir. C'est probablement la manière la plus rapide d'avoir un point de départ. Pour la suite, le programme demandé est trop complexe pour aller bien loin avec un LLM, vous devrez donc prendre en main la bibliothèque pygame pour créer votre interface graphique.

4. EVALUATION

Vous serez évalué.e.s sur via trois notes estimant:

- (1) la qualité du programme réalisé
- (2) la qualité du code et de la documentation (propreté du code, docstring, rapport)
- (3) votre participation au travail réalisé (attention: note individuelle)

Attention: la note totale correspond au minimum des trois notes.

Les scénarios suivants sont donc assurés de vous apporter une mauvaise note:

- Votre groupe a rendu un programme très bon, a rendu une documentation claire et exhaustive mais vous n'avez presque pas participé.
- Votre groupe a rendu un programme très bon, auquel vous avez beaucoup participé, mais le code est sale et mal documenté.

4.1. Programme réalisé. Cette note évalue la quantité et la qualité des fonctionnalités du jeu qui sont implémentées, ainsi que la qualité de l'interface graphique. La liste des fonctionnalités et le barème associé sont divisés en deux tableaux. Le premier tableau liste les fonctionnalités devant être implémentées par tout le monde. Le deuxième tableau liste les fonctionnalités réservées aux étudiants et étudiantes en Syscom, ISI, ROB et MeDH.

Les étudiants et étudiantes en CIMES et IPS doivent viser 20 points pour cette note. Les étudiants des autres parcours doivent viser 30 points (la note sur 30 sera convertie en note sur 20 ensuite). Il n'y aura aucun point bonus accordé aux groupes de CIMES / IPS implémentant des fonctionnalités supplémentaires.

Fonctionnalités principales (pour tout le monde)	Points
Qualité de l'interface graphique et facilité de prise en main	4
Déplacement dans le manoir (section 2.5) et ouverture d'une porte (section 2.6)	3
Tirage et choix des pièces (section 2.7)	3
Gagner des pas avec la nourriture, perdre des pas pendant les déplacements	1
Fin de la partie (gagner en atteignant l'antichambre, perdre si on n'a plus de pas)	1
Ramasser des gemmes, dépenser les gemmes pour choisir une pièce	1
Ramasser des clés, dépenser les clés pour ouvrir les portes	1
Ramasser des dés, dépenser les dés pour tirer à nouveau des pièces	1
Aléatoire dans les objets disponibles dans les pièces	1
Aléatoire dans les pièces tirées au sort et prise en compte de la rareté des pièces	1
Aléatoire dans les niveaux de verrouillage des portes	1
Détecteur de métaux et patte de lapin (et effets associés)	1
Kit de crochetage et ouverture des portes de niveau 1	1

TABLE 1. Barème pour les fonctionnalités principales. Le total du barème donne 20 points. **Tout le monde doit implémenter ces fonctionnalités.**

Le deuxième tableau (à destination des Syscom, ISI, ROB, MeDH) contient des fonctionnalités supplémentaires, qui peuvent potentiellement faire grimper le total de points au dessus de 30.

La note maximale reste de 30/30. Si vous implémentez davantage de fonctionnalités, vous n'aurez pas de points bonus.

Fonctionnalités supplémentaires (Syscom, ISI, ROB, MeDH)	Points
Ramasser des pièces, dépenser les pièces dans les magasins	1
Pelle et possibilité de creuser dans des "endroits où creuser"	1
Marteau et ouverture des coffres	1
Aléatoire dans les objets obtenus dans les coffres, casiers et endroits où creuser	1
Effet spécial d'une pièce qui donne ou retire des ressources lorsqu'on entre dedans (Exemples: Chapel et Bedroom)	1
Effet spécial d'une pièce qui donne ou retire des ressources lorsqu'on tire la pièce (Exemples: Master Bedroom et Weight Room)	1
Effet spécial d'une pièce qui disperse une ressource dans d'autres pièces (Exemples: Patio et Office)	1
Effet spécial d'une pièce qui modifie la probabilité de tirer certaines pièces (Exemples: Greenhouse et Furnace)	1
Effet spécial d'une pièce qui modifie la probabilité de trouver certains objets (Exemples: Veranda et Maid's Chamber)	1
Effet spécial d'une pièce qui ajoute des pièces au catalogue de pièces (Exemples: Chamber of Mirrors et Pool)	1
Un point par autre type d'effet spécial jugé suffisamment différent des autres.	1
Un point par autre type d'objet permanent jugé suffisamment différent des autres.	1

TABLE 2. Barème pour les fonctionnalités supplémentaires du jeu. Ces fonctionnalités sont **réservées groupes de Syscom, ISI, ROB et MeDH**.

4.2. Qualité du code et de la documentation. Le code et la documentation vont être évalués différemment pour les CIMES/IPS, et pour les Syscom, ISI, ROB et MeDH.

Pour les CIMES/IPS, le rapport rendu doit simplement décrire les choix d'implémentations (quelles classes représentent quelles parties du jeu) et leurs justification.

Critère d'évaluation	Points
Clarté et propreté du code (respect des conventions Python)	4
Commentaires et docstrings pour les classes et les méthodes	4
Dépôt Github (requirements.txt, instructions pour installer et lancer le jeu)	4
Bonne utilisation des classes, attributs, méthodes, pour structurer votre code	4
Justification des choix d'implémentation dans le rapport	4

TABLE 3. Barème pour l'évaluation du code et de la documentation **pour les groupes IPS et CIMES**.

Le rapport des Syscom/ISI/ROB/MeDH doit fournir un diagramme de classes UML décrivant le programme. Le diagramme de classes doit être aussi **exhaustif** que possible, et être **cohérent** avec votre implémentation en Python.

Dans le cadre de ce projet, il ne sera pas nécessaire d'implémenter l'encapsulation. Vous pouvez laisser toutes vos méthodes et tous vos attributs publics si vous le souhaitez. Cela n'aura aucun impact sur l'évaluation.

Critère d'évaluation	Points
Clarté et propreté du code (respect des conventions Python)	3
Commentaires et docstrings pour les classes et les méthodes	3
Dépôt Github (requirements.txt, instructions pour installer et lancer le jeu)	2
Structuration du code sous forme de classes	2
Utilisation adéquate de l'héritage (au moins deux classes parentes différentes)	2
Utilisation adéquate de l'abstraction (au moins une classe abstraite)	2
Qualité et exhaustivité du diagramme de classes UML	2
Adéquation entre le diagramme de classes et le code	2
Justification des choix d'implémentation dans le rapport	2

TABLE 4. Barème pour l'évaluation du code et de la documentation **pour les groupes Syscom, ISI, ROB et MeDH.**

4.3. Participation et travail d'équipe. Cette note **est individuelle** et évalue votre investissement dans le groupe, et votre utilisation de git pour travailler en équipe.

Chaque groupe devra créer un dépôt Github public **dès le début** du projet, et y contribuer tout au long du projet. Pour l'évaluation, on regardera si votre dépôt démontre bien votre utilisation régulière de l'outil Git. Vous devez régulièrement déposer vos avancées sur votre dépôt: une bonne habitude est de faire un commit pour chaque fonctionnalité que vous implémentez ou chaque bug que vous corrigez. Tous les membres du groupe doivent participer au dépôt Github. Vous serez pénalisés si:

- vous participez trop peu au dépôt Github
- votre utilisation de git est brouillon (ex: noms de commits inadéquats)
- votre participation au projet est irrégulière (ex: tous vos commits la dernière semaine)

Attention: ce n'est pas une course à qui participe le plus sur le projet. Si vous participez à environ 1/3 du projet (pour un groupe de 3), ce sera considéré comme suffisant. Il n'y aura aucun point bonus pour une personne qui réalise 50% ou même 90% du projet.

Remarque: si vous avez le sentiment que votre groupe vous empêche de participer suffisamment au projet (par exemple: vos collègues implémentent des fonctionnalités que vous deviez implémenter), vous pouvez me contacter par email pour arbitrer la situation.

Pour évaluer votre participation, nous allons aussi nous baser sur votre capacité à présenter le travail à l'oral, et à répondre à des questions sur l'implémentation des fonctionnalités.

Critère d'évaluation	Points
Utilisation adéquate de git, commits réguliers et noms adéquats	8
Présentation orale	6
Réponse aux questions	6

TABLE 5. Barème pour l'évaluation de la participation.