

Projet de Programmation

« Pacman »

Université Paris Est Créteil – Licence 3

27 septembre 2021

1 Le jeu

Le projet consiste en la réalisation d'un jeu s'inspirant du jeu *Pacman* <https://fr.wikipedia.org/wiki/Pac-Man>.

Plus précisément, la partie se déroule sur une grille 2D de cases correspondant à un labyrinthe vu de dessus. Le jeu consiste à déplacer Pacman, un personnage dans un labyrinthe afin de lui faire manger toutes les *pacgomm*es qui s'y trouvent. Quatre fantômes hantent le labyrinthe et s'y déplacent aléatoirement. Si un des fantômes touche le pacman alors le pacman perd une des ses trois vies. En plus des pacgommes classique (bleus), il existe aussi quatre pacgommes bonus (autre couleurs). Ces bonus ont un effet sur le pacman et/ou les *fantômes* et/ou le labyrinthe. Le personnage peut emprunter des passages situés de chaque côté de l'écran, produisant un effet de *wraparound*, le faisant réapparaître de l'autre côté du labyrinthe. Le tableau suivant indique les différents points et effets des pacgommes.

Couleur	Points	Effet
Bleu	100	—
Violet	300	Le pacman devient invisible pour les fantômes. Sa couleur devient jaune pâle.
Orange	500	Le pacman devient un superpacman, sa couleur est alors orange et les fantômes deviennent alors bleus.
Vert	1000	Modifie la structure du labyrinthe

À cela on rajoute les règles suivantes :

- Initialement il a trois vies.
- Si le joueur dépasse les 5000 points, il obtient une vie supplémentaire.
- Chaque fantôme se déplace dans une direction jusqu'à ce qu'il atteigne un mur, puis choisit une nouvelle direction aléatoirement.
- Quand le pacman est invisible, il peut traverser les fantômes sans perdre de vie.
- Quand le pacman est un superpacman, les fantômes deviennent vulnérables. Dans ce cas, ils se déplacent deux fois plus lentement et ils reviennent au centre du labyrinthe s'ils sont touchés par le pacman.
- Le jeu se termine quand il n'y a plus de pacgommes et la partie est gagnée, ou quand le pacman a perdu toutes ses vies et la partie est perdue.

2 Travail demandé

Le projet se déroulera en deux étapes donc deux rendus. Le premier rendu sera un prototype et doit être fait très vite. Le second rendu, la version finale, vous prendra plus de temps et sera attendu pour la fin du semestre.

2.1 Étape 1 : le prototype

Objectif Afin de pouvoir se concentrer sur les aspects fonctionnels du projet, la première étape consistera à réaliser une version du jeu qui fonctionne. Cette version doit contenir toutes les fonctionnalités décrites plus haut. La qualité du code ou de l'architecture ne sera pas jugée pour cette partie. Si tout se passe comme prévu, à la fin de cette première étape, vous disposerez d'une implémentation de *Pacman* qui fonctionne mais dont l'architecture n'est pas optimale voire très mauvaise. Ceci n'est pas grave car le but du prototype est d'avoir une idée précise

de comment implémenter chaque fonctionnalité et de remédier aux bugs naïfs qui peuvent nuire à votre projet. Enfin, ce prototype vous aidera à vous faire une idée globale de toutes les parties constituant le jeu. Cette étape est primordiale pour la suite.

Comment va-t-on juger le prototype ? Le prototype vaut un quart de la note finale du projet (5 points max). Une note maximale sera attribuée à un prototype qui *fonctionne*¹ et dont les deux membres du binôme arrivent à répondre aux questions de manière convaincante. Si vous avez un doute sur la compréhension d'une ou plusieurs des tâches, n'hésitez pas à demander aux chargés de TP, à comparer avec vos collègues, ou même avec le comportement du vrai jeu.

Le prototype est à rendre au plus tard le Samedi 6 Novembre à 23h59 sur EPREL.

Le format du rendu est un zip contenant les fichiers .java. Pour le bon déroulement du projet, veuillez faire attention aux recommandations suivantes :

- Tout autre format sera rejeté et sera équivalent à un prototype non rendu.
- Tout retard est équivalent à un prototype non rendu, prenez vos précautions et arrangez-vous pour soumettre avant la date limite.
- Un prototype non rendu a pour conséquence une note maximale de 15 au projet.

2.2 Étape 2 : le rendu final

Le travail que vous devez effectuer se concentrera sur les aspects liés à la *conception objet*.

En particulier, il faudra que :

- la gestion des états des personnages soit réalisée en utilisant le patron de conception *State*, on pourra trouver une description de ce patron sur Internet ;
- les éléments de la grille de jeu soient hiérarchisés et factorisés convenablement ;
- les aspects logiques du jeu (par exemple la gestion de tous les états internes des personnages et de la grille de jeu) soient indépendants de leur affichage, y compris les couleurs. Vous pouvez utiliser une architecture Observateur/Observé ;
- votre conception permette un ajout facile de nouvelles règles et de nouveaux éléments de terrains.

Le non respect des règles ci-dessous sera considéré comme une *erreur grave* et entraînera un malus allant de 5 à 10 points ;

- le code doit compiler et s'exécuter sans erreur,
- les méthodes doivent être courtes, lisibles même sans commentaire et avoir des noms clairs ;
- l'absence de duplication de code (absence de copier-coller, factorisation),
- les responsabilités de chaque objet (ses méthodes) doivent être clairement identifiées ;
- les objets ne doivent avoir accès qu'aux informations nécessaires pour assumer leurs responsabilités.
- l'utilisation d'images ou de sprites est strictement interdite.

L'accent sera donc mis sur la propreté et la structuration du code et aucunement sur les graphismes ou la diversité des niveaux. En particulier, un niveau avec lequel on peut tester toutes les fonctionnalités est largement suffisant.

La date limite pour rendre vos projets est le 8 janvier 2022

Pour le bon déroulement du projet, veuillez faire attention aux recommandations suivantes :

- Le rendu doit être une archive compressée au format ZIP précisément, aucun autre format ne sera accepté.
- Le nom du fichier doit être de la forme `NOM-PRENOM1_NOM-PRENOM2.zip`.
- Il ne doit contenir **que** les fichiers .java de votre projet organisés en répertoire si nécessaire, et éventuellement des fichiers de description de plateau le cas échéant. Pour cette partie les images et les *sprites* sont **strictement interdits**.
- Vérifiez bien que le contenu de ce fichier ZIP est suffisant pour compiler votre projet en ligne de commande avec `javac` et l'exécuter en ligne de commande avec `java`.
- Tout retard est équivalent à un projet non rendu, prenez vos précautions et arrangez-vous pour soumettre avant la date limite.
- Un projet non rendu a pour conséquence une note maximale de 0.

1. Dont toutes les tâches fonctionnent comme décrit dans l'énoncé du sujet.

La note de la deuxième partie du projet sera entièrement attribuée sur la base d'une *soutenance* (en directe ou enregistrée), durant laquelle chaque groupe présentera le travail réalisé en prenant soin d'expliquer les choix implémentés. Chaque soutenance pourra être suivie d'une séance de questions durant laquelle chaque membre du groupe expliquera certaines parties du code du projet. Chaque membre doit connaître tout le projet et pourra être interrogé sur n'importe quelle partie, y compris les parties qu'il n'a pas réalisées lui-même. Les membres d'un même binôme pourront avoir des notes différentes. On communiquera ultérieurement les modalités exactes de soutenance.

N'hésitez pas à poser des questions, par exemple en envoyant des messages sur Microsoft Teams à vos chargés de TM.