

Devoir Maison « Pac-Man »

L'objectif de ce devoir est de modéliser le plus fidèlement possible une partie de l'architecture du jeu bien connu « Pac-Man », dans lequel le joueur doit réussir à guider le héros éponyme au sein de niveaux successifs pour lui faire gober le maximum de pastilles, et ce tout en évitant les hordes de Fantômes qui y rôdent.



Pour arriver à nos fins, nous allons utiliser certains des diagrammes UML que nous avons étudiés en cours afin de modéliser les éléments « essentiels » revenant obligatoirement lors de la création d'une réplique du jeu « Pac-Man ». A la fin de ce devoir, un développeur maîtrisant un langage orienté objet devrait pouvoir « facilement » créer une réplique du jeu original en s'appuyant sur votre modélisation !

Afin de vous faciliter la tâche, une analyse des besoins a été effectuée ; celle-ci a permis de distinguer plusieurs structures essentielles intervenant dans le jeu « Pac-Man » original. Celles-ci sont détaillées ci-dessous en partant des plus générales, servant à orchestrer l'organisation d'une partie, et en s'achevant sur d'autres plus spécifiques, comme les différents types de personnages.

N'hésitez pas à compléter vos schémas au fur et à mesure de l'énonciation des besoins.

1. Modélisation d'une partie

Nous nous plaçons dans un contexte particulier, à savoir qu'on suppose que le jeu est implémenté sur une borne d'arcade ; plusieurs joueurs peuvent donc se succéder afin d'essayer de faire le meilleur score possible.

La borne d'arcade affiche dans un premier temps un menu simple composé des deux entrées suivantes :

- « New Game », qui permet au joueur de lancer une nouvelle partie.
- « Best Scores », qui affiche la liste des dix meilleurs scores réalisés jusqu'à maintenant.

Il est possible de naviguer dans le menu grâce à deux boutons, ↑ et ↓, et il est ensuite possible de valider son choix grâce à un bouton de validation. Il est possible de quitter le mode « Best Scores » pour revenir au menu principal en appuyant une nouvelle fois sur le bouton de validation.

Via « New Game », le joueur commence une nouvelle partie avec cinq vies, qui permettront à Pac-Man de revivre si celui-ci venait à se faire toucher par un Fantôme, et avec un score initial valant zéro qui augmentera au fur et à mesure de l'avancée de la partie, par exemple en gobant des pastilles ou en terminant des niveaux.

Une partie est constituée d'un niveau actuel, faisant partie d'une liste de trente niveaux prédéfinis au sein de la borne d'arcade. Lorsque le joueur termine le premier niveau, c'est-à-dire lorsque Pac-Man a gobé toutes les pastilles qu'il renfermait, le joueur voit son score augmenter de 500 points et le deuxième niveau est chargé. La partie reprend après une brève pause de quelques secondes permettant au joueur et à la borne de se préparer à la suite des événements. Et ainsi de suite jusqu'à terminer les trente niveaux du jeu ou à arriver au moment où toutes les vies de Pac-Man ont été utilisées.

Dans les deux cas, lorsque la partie est terminée, la borne d'arcade met éventuellement à jour la liste des dix meilleurs scores réalisés jusqu'à maintenant (c'est-à-dire en prenant en compte le score réalisé par le joueur actuel), avant d'afficher une nouvelle fois le menu de sélection initial.

Questions :

- 1.a) Réalisez le diagramme états-transitions d'une borne d'arcade.
- 1.b) Construisez le diagramme états-transitions d'une partie.
- 1.c) Donnez un diagramme de communication correspondant à la situation « le joueur vient de terminer un niveau ».
- 1.d) Créez le diagramme de classes correspondant aux notions de borne et de partie. Veillez à maintenir une certaine cohérence avec les diagrammes réalisés précédemment.

2. Modélisation des niveaux

Maintenant que les entités en charge du déroulement global d'une partie ont été modélisées, nous allons donner plus de consistance à la notion de niveau. Nous définissons un niveau comme étant une matrice composée de 10x10 cases, ces dernières pouvant être de deux natures différentes :

- Les murs, qui sont des cases dans lesquelles les différents personnages (Pac-Man, Fantômes) ne peuvent se déplacer.
- Les autres cases, dites « accessibles », dans lesquelles peuvent se trouver des pastilles et/ou plusieurs personnages (plusieurs Fantômes peuvent se trouver sur une même case à un moment donné).

Une case accessible peut donc éventuellement contenir un objet, qui peut être une pastille ou une super-pastille. Celles-ci sont notamment caractérisées par le nombre de points qui sera accordé au joueur en les gobant : 10 pour une pastille classique, 30 pour une super-pastille.

Questions :

2.a) Créez un diagramme de classes modélisant les notions de niveaux, de cases, et d'objets.

2.b) Complétez le diagramme de la question 1.d en conséquence.

3. Modélisation des personnages

Les niveaux et les cases qu'ils contiennent étant modélisés, nous allons désormais ajouter des personnages à l'intérieur de ceux-ci. De manière traditionnelle, nous dénotons deux types de personnages, à savoir Pac-Man et les Fantômes ; ceux-ci possèdent des caractéristiques communes, parmi lesquelles :

- Ils se situent, à tout moment, sur une case du niveau actuel.
- Ils peuvent se déplacer à chaque intervalle de temps.
- Ils apparaissent chacun sur une case accessible, dite d'apparition, du niveau en cours (une pour Pac-Man, une pour les Fantômes).

La principale différence entre ces deux types de personnages provient de leur comportement sur une case accessible. Notamment, les Fantômes ignorent les pastilles se trouvant éventuellement sur leur case, tandis que Pac-Man, lui, ne se prive pas de les gober. A ce moment, l'objet disparaît de la case et le nombre de points total du joueur est augmenté en conséquence.

En gobant une super-pastille, Pac-Man devient invincible pendant vingt secondes. Lorsqu'il entre dans cet état, les Fantômes ne peuvent plus lui causer de tort : lors d'une collision avec Pac-Man, le Fantôme concerné est détruit et le score du joueur augmente de 100 points. En revanche, si notre héros en touche un sans être invincible, le joueur perd une vie, Pac-Man revient à son point d'apparition, et la partie reprend après une pause permettant, entre autres, de retirer tous les Fantômes restant du niveau (comme cela a été mentionné dans la partie 1).

Les Fantômes ont deux comportements particuliers, dépendant de l'état de Pac-Man :

- Lorsque ce dernier est invincible, les Fantômes du niveau ont plutôt tendance à essayer de l'éviter à tout prix.
- Lorsque ce n'est pas le cas, ceux-ci optent pour une conduite plus agressive et font tout pour essayer d'entrer en contact avec notre héros.

On rappelle que lorsque toutes les vies de Pac-Man sont épuisées, la partie est terminée et les meilleurs scores de la borne sont éventuellement mis à jour avec celui qu'a réalisé le joueur lors de la dernière partie.

D'un point de vue technique, on considère que c'est la partie qui, de manière régulière, s'occupe de déplacer les différents acteurs du niveau et de détecter les différentes collisions qui peuvent arriver (Pac-Man - Fantômes, Pac-Man - pastilles).

Questions :

3.a) Créez le diagramme de communication correspondant à la situation « Pac-Man vient de gober une super-pastille ».

3.b) Construisez le diagramme états-transitions de Pac-Man.

3.c) Donnez un diagramme de communication correspondant à la situation « une collision a lieu entre Pac-Man (dans son état normal) et un Fantôme ».

3.d) Mettez à jour le diagramme de classes de la question 2.b en y incorporant les classes correspondant aux différents personnages.

4. Interaction avec le joueur et collisions

Comme expliqué dans la rubrique précédente, les personnages sont déplacés dans le niveau à intervalles réguliers, mesurés par la borne d'arcade, par l'entité en charge d'orchestrer les différents événements pouvant se produire lors d'une partie.

Les Fantômes ont, comme expliqué précédemment, un comportement dépendant de l'état actuel de Pac-Man : si celui-ci n'est pas invincible, ils vont se rapprocher au maximum de sa position, alors qu'ils vont plutôt tenter de s'éloigner dans le cas contraire.

Pac-Man, lui, avance sans cesse dans la direction lui faisant face. Cette direction est dictée par le joueur au moyen d'un joystick orientable dans quatre directions : lorsqu'il le manipule, la borne détecte la direction concernée et la signale à la partie en cours qui met alors Pac-Man à jour. Puis, lorsque tous les personnages du niveau sont déplacés, Pac-Man avance selon la dernière direction exigée par le joueur, à condition que la case adjacente cible ne soit pas un mur ; dans ce cas, Pac-Man reste sur place et ne change pas de case.

Une fois que la partie a déplacé tous les acteurs du niveau, elle vérifie si des collisions ont lieu. Lorsqu'un tel évènement est détecté, la partie déclenche les actions adéquates et prévient les personnages/objets concernés.

Questions :

4.a) Mettez à jour la classe Pac-Man en prenant en compte les nouvelles informations.

4.b) Donnez un diagramme de communication correspondant à la situation « le joueur appuie sur une touche pour déplacer Pac-Man ».

4.c) Donnez plusieurs diagrammes de communication résumant les différentes actions effectuées de manière régulière par la borne.

4.d) Donnez un diagramme de classes général, par exemple en complétant celui de la question 3.d.

5. Bonus

Si vous êtes arrivé jusqu'ici, c'est que vous avez réussi à modéliser la plupart des comportements de la borne d'arcade et avez trouvé une architecture adéquate pour organiser les différentes entités intervenant dans la version du jeu décrite ici.

Cependant, vous vous êtes peut-être rendu compte que certains points, principalement algorithmiques, n'ont pas forcément été abordés lors de votre modélisation car non représentables par le biais d'UML. Ainsi, certaines réflexions seraient encore nécessaires afin d'implémenter facilement une version du jeu « Pac-Man » en prenant votre modélisation comme base.

Notamment, les deux points suivants mériteraient une analyse approfondie :

- La fréquence d'apparition des Fantômes n'a pas été mentionnée ; on peut, par exemple, imaginer que plus le joueur est proche du trentième niveau et plus l'intervalle séparant l'apparition de deux Fantômes est petit.
- Les déplacements des Fantômes ont été décrits : ils dépendent de l'état de Pac-Man. Il serait intéressant de fournir un algorithme efficace pour faciliter l'implémentation de leur comportement.

Si ces questions vous inspirent (et que votre avancement vous le permet !), vous pouvez essayer d'imaginer des algorithmes pour les traiter. Libre à vous d'explicitier d'autres points qui, d'après vous, mériteraient d'être précisés algorithmiquement !