UNIVERSITÉ LAVAL

Introduction à la programmation

IFT-1004



Travail pratique #4

Un jeu de dames avec interface graphique

(Rapport écrit)

Travail remis à :

M. Pascal GERMAIN

Travail effectué par :

Ariane Fiset (111 236 136)

Pascal de Le Rue (111 187 065)

Session été 2020

Remis le 26 juillet 2020

# 4.1 Fonctionnalités obligatoires

Lors de la première partie du travail pratique, nous avons fait plusieurs choix afin d’obtenir un jeu fonctionnel en tout temps. De plus, ce jeu comporte des fonctionnalités qui facilite son utilisation, tel que des boutons ou des zones de textes. Les fonctionnalités du code ont été fait de la façon suivante :

(1) Afin que les pièces puissent se déplacer sur le damier, nous avons tout d’abord initialisé trois variables, soit un booléen afin de savoir si une pièce est sélectionner ou non, une variable de pièce sélectionné et une de position sélectionnée. On a ensuite trouvé le numéro de ligne et de colonne associé à cette position pour ensuite déplacer la pièce à la position sélectionner et finalement actualiser l’affichage du damier. (2) Afin de bloquer les déplacements invalides, nous avons tout simplement fait appel à la classe *Partie* afin de pouvoir réutiliser les ‘règlement’ déjà codés. (3) Encore une fois, nous avons utilisé la classe *Partie* afin que le joueur puisse choisir une nouvelle pièce source après un mouvement invalide. (4) Afin qu’une prise soit obligatoirement fait si elle est possible, nous . (5) Pour que les prises multiples soient gérées correctement, (6) L’alternance du tour des joueurs a été réalisé. (7) Un bouton *Nouvelle partie* a été ajouter à l’interface à l’aide de la classe Button de tkinter. Lorsqu’un utilisateur appuie sur ce bouton, la fonction *nouvelle\_partie* est lancé et la classe *FenetrePartie* est supprimé et ensuite initialiser afin de redémarrer une nouvelle partie. (8) Un bouton *Quitter* est aussi ajouté à l’interface afin de permettre à l’utilisateur de Quitter la partie. Ce bouton ne fait qu’effectuer la commande *quit* ce qui revient à la même chose qu’un clic sur le *X* de la page. (9) Une rétroaction est donnée à l’utilisateur en tout temps. Cette fonctionnalité est effectuée, car tous les déplacements découlent de la classe *Partie* qui effectuait déjà ce type de messages. Ces messages sont donc maintenant transmis à l’interface graphique du jeu à l’aide du module *Label*. (10)

Affiche la couleur du joueur qui doit faire un déplacement (tour courant) : Cette affichage a été créer un peu de la même façon que la rétroaction, mais, cette fois-ci,

Finalement, nous avons aussi garder dans notre code la représentation du jeu dans la console. Ceci nous a permis de visualiser ce qui se passait en temps et lieux dans le code afin de synchroniser parfaitement l’interface graphique avec le jeu.

# 4.2 Fonctionnalités optionnelles

Dans ce travail pratique, nous avons choisi les fonctionnalités ainsi que la façon de les résoudre suivante :