

Contexte

Pour ce projet, nous voulons réaliser le jeu du Yam's (ou Yahtzee). Il s'agit d'un jeu de dés pouvant être joué à deux joueurs ou plus. Ce projet se consacrera uniquement sur les règles du jeu de **Yahtzee** *Domaine Public* (wiki : https://fr.wikipedia.org/wiki/Yahtzee#Yahtzee_Domaine_public).

1 Objectif

L'objectif de ce projet est de réaliser un programme qui permet de jouer à ce jeu et qui en appliquera toutes les règles. Ainsi, les joueurs n'auront plus à se prendre la tête à calculer les points et n'auront qu'à réfléchir à la meilleure stratégie pour enfler tout le monde !

2 Préparation de nos classes

Nous travaillons dans le langage C++ où tout est objet. A minima, les classes suivantes devront apparaître dans votre programme :

- Une classe abstraite **figure** qui servira de référence à toutes les figures.
- Une classe par figure possible (ex : brelan, full, carré, etc.). Si une figure peut avoir plusieurs valeurs différentes comme un brelan de "2" ou de "3" alors cette figure doit être une classe template.
- Un **lancer** qui représente un lancer de dés.
- Une classe **joueur** qui gère les points d'un joueur.

Vous avez tout le loisir d'utiliser l'entièreté du standard C++ (<https://en.cppreference.com/w/>) et mettre en avant les concepts vus en cours. Enfin, si un tri est nécessaire, celui-ci **doit** se faire en utilisant la fonction `std::sort`.

3 Conclusion

Il paraît évident qu'à la lecture de ce sujet, vous vous soyez dit qu'il fallait faire les tests et vérifications permettant de valider le bon fonctionnement des différentes mécaniques du jeu. Un vrai plus majeur serait également de réaliser une petite vidéo ou des images de votre réalisation pour apprécier votre solution ! Empaqueter votre projet dans un fichier zip contenant tout ce qu'il faut. N'oubliez pas de mettre vos noms dans votre répertoire de soumission et dans l'en-tête de votre fichier `lisez-moi.txt`.

4 Bonus

Cette section liste les éléments supplémentaires (liste non-exhaustive) pouvant jouer en votre faveur. Cependant, la faveur n'est accordée que si les fonctionnalités de bases sont présentes. Notez également que si votre version du jeu est solide, nombreuses d'entre elles peuvent être ajoutées assez facilement.

- Pouvoir jouer à différentes variantes (qui pourraient servir de niveau de difficulté), exemple :
 - facile : il est possible de faire les figures dans l'ordre que l'on souhaite.
 - normal : la partie supérieur doit être faite avant la partie inférieur.
 - difficile : la figure doivent être faites dans l'ordre dans lequel elles sont données.
 - hard-core : même chose que la version difficile, mais en plus, les figures sont ordonnées aléatoirement.
- Un système de sauvegarde, afin de mettre la partie en pause et reprendre un autre jour.
- Faire des IA basiques. Par exemple, une qui essaiera toujours de maximiser ses points au moment de jouer, sans avoir de stratégie sur le long terme. Une qui prend des décisions sans queue, ni tête. Bref, libre à vous !
- Si la console vous ennuie, rendez ça plus visuel (par exemple : <https://www.sfml-dev.org/>).
- Une couche réseau pour jouer à distance (voir la partie réseau de sfml par exemple).
- Libre à vous de faire plus encore en le mentionnant dans votre fichier `lisez-moi.txt` !