

## TP 6 : Jeu de Yams

### WIMS.

20 minutes de contrôle sur WIMS.

### Exercice 1 (Jeu de Yams : le TD en pratique).

Pour ce TP nous allons travailler en parallèle dans l'application web `Jupyter` et dans un fichier source `squelette.cpp` afin, à la fin du TP, d'obtenir un exécutable autonome permettant de jouer une partie de Yams simplifié.

Comme d'habitude, téléchargez et décompressez l'archive `Semaine6` grâce aux commandes vues aux TP précédents.

Lancez l'application `Jupyter` depuis le terminal avec la commande suivante :

```
jupyter3-notebook
```

Cette application ouvre une fenêtre dans votre navigateur web. Naviguer jusqu'à votre répertoire `Info111/Semaine6`.

Ouvrez la feuille `tp6-feuille-1.ipynb`, elle contient l'essentiel du TP.

Lorsqu'il vous est demandé de compiler du code à la main, **ouvrez un nouveau terminal pour cela**, faites attention à ne pas quitter `Jupyter` !

### Exercice ♣ 2 (Quelques extensions).

Le programme que vous obtenez à la fin de l'exercice 1 est encore loin de représenter le jeu de Yams réel. Dans cet exercice nous vous proposons plusieurs extensions déjà évoquées dans le sujet de TD qui vous permettrons de vous en approcher.

Vous pouvez utiliser `Jupyter` pour prototyper vos fonctions, mais vous devez au final implémenter ces extensions dans votre programme compilé (fichier `yams.cpp`).

- Dans le vrai jeu de Yam's, le joueur peut relancer jusqu'à trois fois un ou plusieurs dés avant de choisir une figure.
  - (1) Ajouter une fonction `vector<int> relance(int nde, vector<int> des)` qui "relance" le dé choisi en premier argument et le remplace donc par un nouveau nombre aléatoire entre 1 et 6.
  - (2) Dans la boucle de jeu, ajouter les instructions nécessaires pour que le joueur puisse choisir jusqu'à trois dés à relancer et les relancer.
- La partie de Yams se termine lorsqu'un joueur a marqué des points pour toutes les figures possibles.
  - (1) Ajouter dans le `main` un tableau de score contenant une case pour chaque figure.
  - (2) Lorsque le joueur choisit une figure, les points qu'il gagne doivent être stockés dans la partie correspondante du tableau. Une fois une case du tableau remplie, elle ne peut plus être modifiée.
  - (3) La partie se termine lorsque toutes les cases du tableau sont remplies. Le score du joueur correspond à la somme des cases du tableau.

- C'est un peu triste de jouer tout seul! Modifiez la boucle principale pour alterner les tours entre 2 joueurs dont les scores seront stockés dans deux tableaux de score distincts. La partie s'arrête lorsqu'un joueur a fini de remplir son tableau de score. On compare alors le score de chaque joueur pour désigner le gagnant.
- Si vous êtes très motivés vous pouvez transformer une dernière fois votre programme afin de rendre possible des parties entre un nombre quelconque de joueurs. Pour cela vous aurez besoin d'utiliser un tableau de tableaux de scores. Ce tableau contiendra un tableau de score pour chaque joueur présent dans la partie.