

# Introduction à la programmation C : Projet Implémentation d'un jeu de Yahtzee

Licence Informatique 2ème année  
Université de Nice Sophia Antipolis

## Organisation et Planning

Le projet doit être réalisé individuellement.

Vous devez implémenter en C le projet tel que décrit ci-dessous et rendre avant le 13 Mai 2011 à 8h00 :

1. l'ensemble du code source du projet (fichier source, makefile + notice expliquant comment créer l'exécutable et précisant sur quel OS votre projet fonctionne).
2. 4 pages (maximum) expliquant comment vous avez réalisé ce projet (descriptif des fonctions, organisation du programme ...)

La note tiendra particulièrement compte de l'absence de bug et la cohérence de l'ensemble.

## Le projet : implémenter un jeu de Yahtzee

### Le Yahtzee : les règles du jeu

Chaque joueur lance les dés lorsque vient son tour. Son but est, à l'aide des cinq dés, de réaliser une figure (5 dés identiques, une paire, un triplet...). Pour réaliser cette figure, il a le droit à trois jets de dés par tour et il est à chaque jet libre de les relancer tous ou juste ceux de son choix.

Le gagnant est celui qui a fait le plus de points, certaines figures donnant des points supplémentaires si elles sont réussies.

Chaque joueur dispose d'une feuille de score sur laquelle est inscrit la liste des figures qu'il doit réaliser. À chaque tour, une case doit nécessairement être remplie avec le score correspondant. Par exemple le résultat au dé : (3,3,3,2,2) peut être inscrit dans les deux avec 4 points, dans les trois 9 points, full avec 25 points, etc. On peut être obligé ou choisir par tactique de remplir une case qui ne convient pas (avec le jet précédent, la case '5 dés identiques' par exemple), on y met alors le score 0. On ne peut plus ensuite revenir sur une case remplie, chaque combinaison ne peut donc être réalisée qu'une fois.

La feuille de score est divisée en deux parties, organisée comme suit :

## Partie supérieure

Combinaison	Points
As	1 × le nombre de dés 1 obtenus
Deux	2 × le nombre de dés 2 obtenus
Trois	3 × le nombre de dés 3 obtenus
Quatre	4 × le nombre de dés 4 obtenus
Cinq	5 × le nombre de dés 5 obtenus
Six	6 × le nombre de dés 6 obtenus
<b>Sous-total</b>	<b>Somme des points obtenus ci-dessus</b>
Prime	35 points si le sous-total est supérieur ou égal à 63 points
<b>Total I</b>	<b>Somme du sous-total et de l'éventuelle prime</b>

La prime s'obtient en ayant au moins 63 points. Pour obtenir 63 points, il faut avoir obtenu trois as, trois deux, trois trois, etc. Mais rien n'empêche de « rater » un deux, par exemple, puis de le rattraper en faisant quatre trois ou quatre quatre...

## Partie inférieure

Combinaison	Description	Points
Brelan	Trois dés identiques	Somme de tous les dés
Petite suite	1,2,3,4 ; 2,3,4,5 ou 3,4,5,6	15 points
Grande suite	1,2,3,4,5 ou 2,3,4,5,6	20 points
Full	Trois dés identiques + deux dés identiques	30 points
Carré	Quatre dés identiques	Somme de tous les dés + 20 points
Yam's	Cinq dés identiques	50
<b>Total II</b>	<b>Somme des points obtenus ci-dessus</b>	

Le score d'un joueur est la somme **Total I** + **Total II**.

## Travail à réaliser

Vous devez implémenter le jeu du Yahtzee tel que décrit dans la section précédente.

En début de partie, vous devez demander combien de joueurs participent. Chaque joueur joue à tour de rôle, on parle de mode *hot seat*.

Le tour d'un joueur se compose d'un premier lancé de dés, il choisit alors les dés à relancer une première fois puis une deuxième fois. A la suite il sélectionne la combinaison qu'il valide, éventuellement une qui ne lui apporte aucun point. A n'importe quelle étape un joueur peut décider de cesser de relancer les dés et valider de suite une combinaison. Une fois cela fait, on passe au joueur suivant jusqu'à ce que le tableau des scores de tous les participants soient remplis.

En fin de partie, un récapitulatif des scores devra être affiché avec le nom du vainqueur. Les dix meilleurs score sur l'ensemble des parties jouées seront sauvegardés dans un fichier.

On doit pouvoir interrompre une partie et la sauvegarder pour la reprendre ultérieurement. Le menu devra donc comporter les options "nouvelle partie", "sauvegarder la partie", "charger une partie en cours" et "afficher les meilleurs scores".

## Le Rapport

En même temps que le projet, vous devez remettre un rapport. Celui devra être court, quatre pages au plus. Vous devrez mettre en avant les fonctions principales que vous avez utilisées et

expliquer les choix d'implémentation, votre démarche. Il comptera dans la notation.

De même, il est vivement conseillé de commenter les parties clés de votre code et d'adopter un style de codage clair et cohérent. Préférez les fonctions courtes avec un rôle précis. Gérez les cas exceptionnels, les entrées erronées de l'utilisateur ne doivent pas interrompre le jeu.

## Note sur l'aléatoire en C

En C, on peut utiliser la fonction `rand()` de la librairie `stdlib.h` pour obtenir un entier aléatoire entre 0 et `RAND_MAX`. Les nombres obtenus dépendent d'une valeur initiale appelée graine, il faut donc changer cette valeur à chaque exécution du programme pour avoir des parties différentes. Cela peut se faire via l'appel suivant

```
srand(time(NULL));
```

## Aller plus loin

Toute initiative personnelle (et justifiée) sera prise en compte et valorisée. Il existe de nombreux moyens d'étendre le projet. Une possibilité appréciable est d'autoriser des joueurs ordinaires dans les parties. Cela revient à implémenter une "intelligence artificielle" simple dans le projet.

## Les choses à faire et à ne pas faire

- **A faire**
  - Appeler `srand` au plus une fois dans le programme
  - Libérer la mémoire allouée avec `malloc` par des `free`
  - Vider le tampon de lecture quand nécessaire
  - Tester votre logiciel
- **A ne pas faire**
  - Utiliser des variables globales en dehors de toute fonction
  - Appeler `fflush(stdin)`
  - Récupérer des chaînes de caractère en utilisant `gets` ou `scanf`, utiliser à la place `fgets` ou des appels successifs à `fgetc`