

Guide de Développeur:

Description du programme:

Le programme réalisé en C a pour objectif de simuler un tournoi sportif dont les équipes peuvent être choisies par l'utilisateur (un nombre d'équipes en puissance de 2). Ici, le sport sélectionné a été le tir à l'arc entre 2 équipes sur une cible.

Les équipes lancent 3 fois une flèche sur une cible et remportent des points entre 0 et 100. Si le match est nul, alors le match continue. Ceci est fait pour tous les matchs du tournoi. Les matchs durant un tournoi sont réalisés simultanément à l'aide de threads.

À la fin du tournoi, le vainqueur est affiché. Une fois l'exécution terminée, vous pourrez retrouver l'historique du tournoi dans un fichier créé par le programme nommé "resultat.txt".

L'algorithme du programme:

Le programme prend un fichier qui contient des équipes dont le nombre est une puissance de deux.

Le tournoi utilise un système d'élimination des joueurs perdants pour chaque tour jusqu'à ce que le vainqueur du tournoi reste à la fin. Pour chaque round sont éliminés la moitié des équipes.

Les Structures:

Le programme réalisé utilise 2 structure que sont les suivantes :

-Une Structure Équipe qui permet de représenter une équipes dans le tournois, elle est caractérisé par :

- nom : chaîne de caractères représentant le nom de l'équipe.
- score : Entier qui représente le score de l'équipe.
- continuePlaying : Booléen qui permet de savoir si l'équipe peut continuer à faire des matchs dans les prochains tours.

-Une structure Match qui permet de représenter un match entre deux équipes, elle est caractérisé par :

- eq1 : Pointeur vers la première équipe
- eq2 : Pointeur vers la deuxième équipe
- round : Entier indiquant le tour dans lequel le match est joué

Les Fonctions

Le programme réalisé nécessite l'utilise des fonctions suivantes :

- readFile:

Prend en entrée le nom d'un fichier texte contenant les équipes et renvoie un tableau de chaînes de caractères qui contient les noms d'équipes. Elle utilise la fonction malloc pour allouer de la mémoire pour le tableau.

- melange:

Prend en entrée un tableau d'équipes et un entier qui représente le nombre d'équipes. La fonction rand pour générer des nombres aléatoires est utilisée pour réaliser un mélange.

- JouerUnMatch:

Prend en entrée un pointeur vers une structure Match et permet de réaliser un match. Cette fonction nécessite l'utilisation d'un mutex pour synchroniser l'accès aux équipes pendant un match.

- estPuissanceDe2:

Prend en entrée un entier et renvoie un booléen indiquant si l'entier est une puissance de 2.

- checkArgs:

Prend en entrée le nombre d'arguments passés au programme et renvoie un entier représentant le nombre d'arguments passés en paramètres.

- Verif_Result_File:

Elle vérifie si le fichier "resultatT.txt" existe. Si ce fichier n'existe pas, alors elle est créée.

- WriteResult:

Prends en entrée un texte et écris ce texte dans le fichier "resultatT.txt". La fonction utilise un mutex pour synchroniser l'accès au fichier.

Fonctionnalité futur à ajouter:

- Améliorer l'affichage des résultats
- Permettre à l'utilisateur d'intervenir durant les matchs en sélectionnant un vainqueur de match
- Ajouter une interface graphique

Répartition des Tâches :

Je me nomme GUNDUZ Maxime et j'ai dû travailler entièrement seul sur ce projet, de ce fait j'ai réalisé toutes les tâches de ce projet (implication a 100% pour toutes les tâches pour GUNDUZ Maxime).