

# RAPPORT EXPLICATIF DU CODE

## I. Introduction

Ce programme fait en C est un jeu de dactylographie qui permet aux utilisateurs de taper des phrases le plus rapidement possible. Il est composé de plusieurs fonctions pour démarrer le jeu, configurer les phrases, afficher une aide et gérer le menu principal.

## II. Fonctionnalités principales

- Affichage des noms des participants
- Affichage d'une aide depuis un fichier texte
- Configuration des phrases à taper
- Jeu de dactylographie
- Menu principal pour naviguer entre les options

## III. Explications détaillées du fonctionnement

- Bibliothèques et définition des macros inclues

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <string.h>
```

Ces bibliothèques sont incluses pour gérer les entrées/sorties, l'allocation dynamique de mémoire, les fonctions de temps et de chaîne de caractères.

```
#define ANSI_COLOR_RESET "\033[0m"
#define ANSI_COLOR_GREEN "\033[0;32m"
#define ANSI_COLOR_RED "\033[0;31m"
#define ANSI_COLOR_YELLOW "\033[0;33m"
#define ANSI_COLOR_BLUE "\033[0;34m"
#define ANSI_COLOR_CYAN "\033[0;36m"
```

Ces macros définissent les codes d'échappement ANSI pour la coloration du texte dans le terminal, ce qui améliore la lisibilité et l'interactivité du programme.

- Structure **Joueur**

```
typedef struct {
    char nom[50];
    double temps;
} Joueur;
```

Cette structure permet de stocker le nom du joueur et son temps de saisie.

- Fonction **start**

```
void start() {
    ...
}
```

Cette fonction affiche les noms des participants avec une pause entre chaque affichage pour un effet visuel agréable, puis efface l'écran.

➤ Fonction **afficherAide**

```
void afficherAide() {
    ...
}
```

Cette fonction ouvre un fichier 'aide.txt' et affiche son contenu. Elle utilise une boucle 'while' pour lire chaque caractère jusqu'à la fin du fichier (EOF).

➤ Fonction 'configurer'

```
void configurer() {
    ...
}
```

Cette fonction permet à l'utilisateur d'ajouter des phrases dans un fichier 'phrases.txt'. Elle utilise une boucle 'for' pour lire plusieurs phrases et les écrire dans le fichier.

➤ Fonction jeu

```
int jeu() {
    ...
}
```

Cette fonction est le cœur du jeu. Elle suit les étapes suivantes :

- Lecture des phrases depuis le fichier :
  - Utilise une boucle **while** pour lire chaque phrase et les stocker dans un tableau dynamique.
- Saisie des noms des joueurs :
  - Utilise une boucle **for** pour demander les noms des joueurs.
- Déroulement du jeu :
  - Pour chaque joueur, une phrase est sélectionnée aléatoirement, et le joueur doit la taper. Le temps de saisie est mesuré à l'aide de **time**
- Tri des résultats :
  - Les joueurs sont triés par leur temps de saisie en utilisant le tri à bulles (boucle **for** imbriquée).
- Affichage des scores :
  - Affiche les temps de chaque joueur en ordre croissant.

➤ Fonction **main**

```
int main() {
    ...
}
```

Cette fonction affiche le menu principal et permet à l'utilisateur de choisir entre démarrer le jeu, configurer les phrases, afficher l'aide ou quitter le programme. Elle utilise une boucle **do-while** pour permettre la navigation dans le menu jusqu'à ce que l'utilisateur choisisse de quitter.

➤ Utilisation des boucles, des conditions et autres

Boucles for :

- Utilisées pour itérer sur les participants dans la fonction **start**
- Pour lire plusieurs phrases dans la fonction **configurer**
- Pour gérer les joueurs et les résultats dans la fonction **jeu**

Boucles while :

- Utilisées pour lire le fichier d'aide jusqu'à la fin dans **afficherAide**
- Pour lire les phrases depuis le fichier dans **jeu**
- Pour vérifier si la phrase saisie par le joueur est correcte dans **jeu**

Conditions if else :

- Vérification des erreurs lors de l'ouverture des fichiers dans **afficherAide** et **configurer**
- Vérification de la saisie correcte de la phrase dans **jeu**
- Gestion des choix du menu principal dans **main**

➤ Allocation dynamique de mémoire :

- Utilisée pour stocker les phrases lues depuis le fichier dans **jeu**

➤ Génération de nombres aléatoires :

- Utilisée pour sélectionner une phrase aléatoire dans **jeu**

➤ Gestion des couleurs :

- Utilisation des macros définies pour colorer le texte, rendant l'interface utilisateur plus attrayante

Ce programme combine diverses fonctionnalités pour offrir une expérience de jeu interactive et éducative. Les boucles, conditions et autres constructions de programmation sont utilisées de manière efficace pour gérer les entrées/sorties, le traitement des données et l'interaction utilisateur. Les fonctionnalités comme la configuration dynamique des phrases et l'affichage coloré du texte ajoutent à la convivialité et à la flexibilité du programme.