

TD-TP n° 10 Programmation Impérative Fichiers – Compilation séparée - Arguments de la ligne de commandes

Exercice

Reprendre l'exercice 1 de la partie TP7, et rajouter comme fonctionnalité à votre programme, le fait de pouvoir stocker les notes saisies dans un fichier, et de pouvoir également relire les notes stockées dans ce fichier.

- 1) Dans une première version vous écrirez le nombre de notes de votre tableau sur la première ligne du fichier, puis les notes séparées par des tabulations sur la 2^{ème} ligne (vous relirez les notes à partir du fichier selon le même format)
- 2) Dans une 2^{ème} version, vous écrirez/lirez en bloc les notes du tableau

Remarque : le `fread` et le `fwrite` ne peuvent pas écrire une seule donnée, et ne fonctionnent qu'en mode binaire, alors que le `fprintf` et `fscanf` ne fonctionnent qu'en mode texte

Pour résoudre le problème vous stockerez le nombre de notes à lire ou écrire dans un fichier texte séparé du fichier binaire contenant les notes.

- 3) Reprendre le programme obtenu à l'issue de la 1^{ère} question.

Vous diviserez le code en trois fichiers `main.c`, `tableau.c`, `fichier.c`, qui contiendront respectivement la fonction `main`, les fonctions relatives à la gestion du tableau, les fonctions relatives aux opérations d'écriture et de lecture des fichiers.

Les fichiers `tableau.c` et `fichier.c` seront associés à un fichier `tableau.h` et `fichier.h`.

Modifier le programme de telle façon que les noms des fichiers à traiter puissent être passés en arguments par l'interpréteur de commandes :

```
nomExecutable fichierEntree
```

L'argument `fichierEntree` est obligatoire et correspond au nom du fichier qui est à traiter en entrée.