



Outils de développement logiciel

TP n° 4 : règles de codage et production de documentation

Alain Lebreton

2023-2024

Objectif

Mettre en oeuvre les règles de codage et produire de la documentation.

Pré-requis : chapitre 8

Durée estimée : 1 séance

1 Mise en forme du code

Exercice n° 1 : conversion degrés Fahrenheit / Celsius

La conversion des degrés Fahrenheit en degrés Celsius consiste à prendre la température en degrés Fahrenheit à laquelle on soustrait 32. Puis, on divise le résultat par $\frac{9}{5}$ (1, 8). Par exemple pour 50 °F, on obtient : $50 - 32 = 18$, puis $\frac{18}{1,8} = 10$, donc 50 °F = 10 °C. Dans cette unité et sous pression atmosphérique normale, l'eau pure gèle à 32 °F exactement et bout à 212 °F. La table doit afficher les différentes valeurs comprises entre 0 et 100 degrés Fahrenheit, par pas de 10 degrés.

1. Déplacez-vous dans votre dossier "odl" et créez le sous-dossier "tp04", puis placez-vous dans ce dernier.
2. Copiez dans le sous-dossier "fahr2cels" que vous aurez créé dans "tp04", le fichier "fahrenheit2celsius.c" que vous trouverez dans le dossier "ressources-odl-fise/tp04/", puis placez-vous dedans.
3. Compilez "fahrenheit2celsius.c" :

```
$ gcc fahrenheit2celsius.c -o fahrenheit2celsius
```

S'il n'y a pas d'erreurs de compilation exécutez le programme :

```
$ ./fahrenheit2celsius
```

Que se passe-t-il ?

4. À l'aide de **Vim**, formatez correctement le fichier, corrigez les éventuelles erreurs dans le code et dans les commentaires.

Exercice n° 2 : matrice d'acquiescement

La matrice d'acquiescement est un exemple simple de jeu mathématique. Soit la matrice d'acquiescement suivante :

$$\begin{pmatrix} 1 & -2 \\ 0 & 1 \end{pmatrix}$$

Le joueur **L** choisit une ligne au hasard, le joueur **C** choisit une colonne lui aussi au hasard. Ils annoncent leur choix simultanément. Le nombre qui est à l'intersection de cette ligne et de cette colonne devient alors la somme que **C** doit payer à **L**. Puis on passe au coup suivant. Par exemple, si **L** choisit la ligne 1 et **C** la colonne 1, alors **C** donne 1 allumette à **L**. Par contre, si **L** choisit la ligne 1 et **C** la colonne 2, **L** donne 2 allumettes à **C**, à cause du signe '-'. Ici, l'ordinateur prend la place de **L**.

1. Copiez dans le sous-dossier "game" que vous aurez créé dans 'tp04', le fichier "zero_sum_game.c" que vous trouverez dans le dossier "ressources-odl-fise/tp04/", puis placez-vous dans "game".
2. Compilez "zero_sum_game.c" de la manière suivante :

```
$ gcc zero_sum_game.c -o zero_sum_game
```

S'il n'y a pas d'erreurs de compilation exécutez le programme :

```
$ ./zero_sum_game
```

Que se passe-t-il ?

3. À l'aide de **Vim**, formatez correctement le fichier, corrigez les éventuelles erreurs dans le code et dans les commentaires.

2 Production de la documentation

Exercice n° 3 : mettre en oeuvre Doxygen

1. Dans le dossier "fahr2cels", éditez avec **Vim** le fichier de configuration "fahr2cel_conf" permettant de produire la documentation du programme "fahrenheit2celsius" à l'aide de l'outil **Doxygen**. La documentation sera générée dans un sous-dossier "doc".
2. Même chose pour le fichier de configuration "game_conf" du programme "zero_sum_game".

3 Livrable

Avant la date limite, déposez sur la plateforme Moodle une archive compressée sous le nom "rapport-tp04-nom2-nom1.tgz". Cette archive inclura un fichier "README.md", les deux fichiers C corrigés, ainsi que les fichiers de configuration de **Doxygen**. Le fichier "README.md" indiquera votre nom, la description des fichiers de l'archive, ainsi que l'explication permettant de compiler et de produire la documentation.

4 Résumé

Dans ce TP, vous avez remis en forme des programmes C "mal formés" et produit des documentations à l'aide de l'outil **Doxygen**.