

## Rapport TP3 Prog C

Ce rapport contient un compte rendu des questions posées lors du TP ainsi qu'un annexe pour pouvoir compiler les différents exercices.

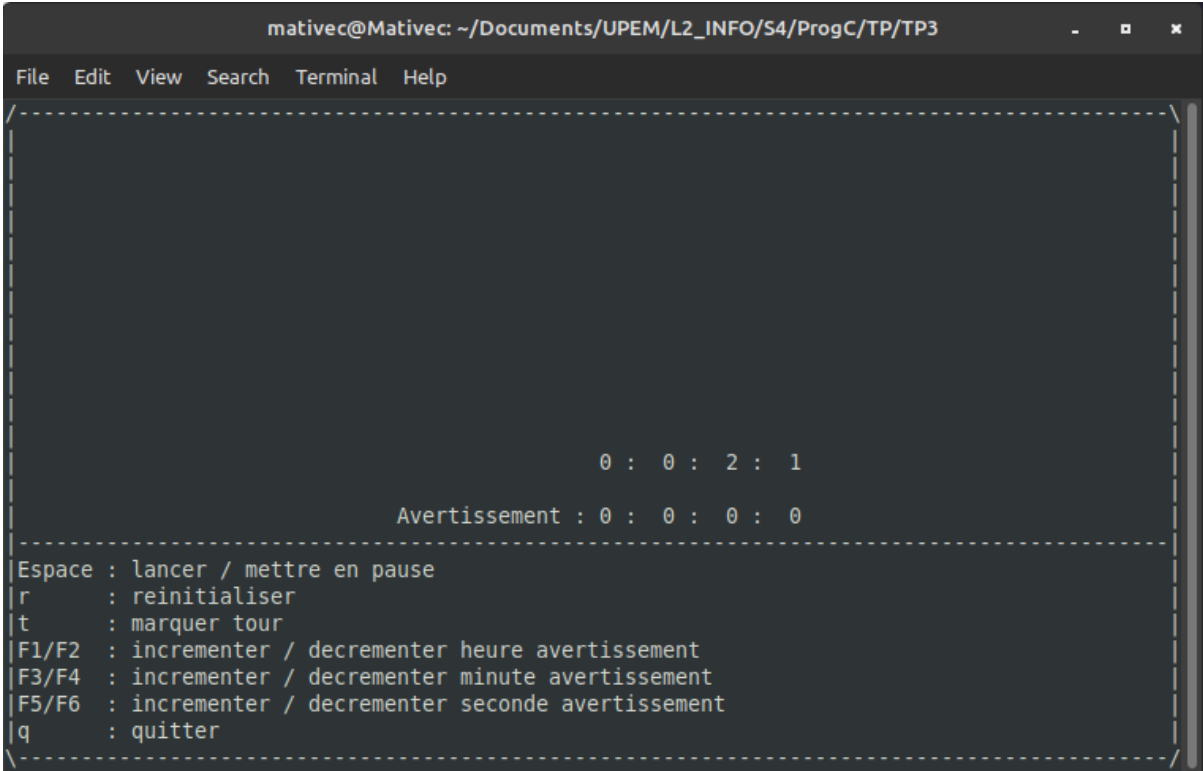
### Introduction:

Le but de ce tp est de réaliser un chronomètre le plus "complet" possible avec un affichage avec ncurses, avec des fonctionnalités basique d'un chronomètre.

### Guides d'utilisation:

Cette partie est pour la dernière version du TP ou un affichage graphique à été réalisé avec le module ncurses.

Voici comment ce présente le programme:



```
mativec@Mativec: ~/Documents/UPEM/L2_INFO/S4/ProgC/TP/TP3
File Edit View Search Terminal Help
-----
                                0 : 0 : 2 : 1
                    Avertissement : 0 : 0 : 0 : 0
-----
Espace : lancer / mettre en pause
r       : reinitialiser
t       : marquer tour
F1/F2   : incrementer / decrements heure avertissement
F3/F4   : incrementer / decrements minute avertissement
F5/F6   : incrementer / decrements seconde avertissement
q       : quitter
-----
```

Plusieurs options s'offrent à vous, comme indiqué. Vous pouvez choisir de lancer le chronomètre directement réaliser un avertissement avec le temps

voulu ils vous suffit juste d'appuyer sur les touches ou quand ce dernier est atteint une "alarme" est affiché pour prévenir que le temps est dépassé.

Vous pouvez aussi réaliser des tours comme pour un chronomètre basique pour enregistrer différents temps souhaités.

Voilà toutes les options implémenter lors de ce TP. Ce dernier peut bien sûr être amélioré, mais toutes les fonctions de base d'un chronomètre ont été réaliser.

## **Questions TP:**

### **Exercice 1:**

Dans un premier temps nous avons réalisé dans cet exercice un chronomètre très basique ou ce dernier se finit avec un CTRL+c, ce qui n'est pas vraiment une bonne chose. L'affichage se fait avec la sortie standard dans cette exercice, le but étant de réaliser les fonctions permettant de réaliser le dis chronomètre l'affichage étant la pour tester si le chronomètre fonctionne.

Différentes fonctions ont été réalisées lors de cet exercice, et vont être utilisées pour les prochaines parties, ces dernières seront utilisées pour calculer le temps.

Nous avons 5 fonctions différentes:

- *int intervalle\_ms(struct timeval debut, struct timeval fin);*  
qui renvoie la différence de temps en milliseconde.

-*int nb\_ms\_vers\_centiemes(int nb\_ms);*  
pour passer d'un temps en milliseconde, en centièmes de seconde

-*int nb\_ms\_vers\_secondes(int nb\_ms);*  
pour passer d'un temps en milliseconde, en seconde

*-int nb\_ms\_vers\_minutes(int nb\_ms);*

pour passer d'un temps en milliseconde, en minutes

*-int nb\_ms\_vers\_heures(int nb\_ms);*

pour passer d'un temps en milliseconde, en heures

### Exercice 2:

Pour cette exercice, le but est de réutiliser les fonctions déjà implémentées, lors de l'exercice précédent et de commencer à gérer un affichage basique avec ncurses et la gestion des boutons pour lancer et arrêter le chronomètre, ainsi que pour quitter le programme de manière plus "propre".

### Exercice 3:

Pour cette dernière partie on réalise un "vrai" chronomètre avec un affichage clair et concis. Pour cette exercice l'ensemble des fonctions, programmes réalisés ont été réutilisés pour cette dernière partie.

Différentes fonctions ont été réalisées, qui sont toutes des fonctions d'affichage pour la plupart. Une structure chronomètre prenant les paramètres demandés a été réalisée ainsi qu'un fichier .h et .c ou toutes les fonctions utiles ont été utilisées.

Le guide d'utilisation plus décrit déjà le fonctionnement du programme pour plus de détaille consulter le programmes donné ci-joint.

### Conclusion:

Le TP dans son ensemble c'est bien passé, les difficultés rencontrées lors du dernier TP à savoir la gestion des couleurs avec le modules ncurses ont été résolus et la compréhension de ce dernier est acquise. Les difficultés rencontrées ont été la compréhension de l'exercice 4 (cryptage césar). Si la réalisation du code n'as pas été difficile, la compréhension du fonctionnement du modulo (dans le cas de l'exercice). Notamment de passer d'un chiffre à une lettre.

## Annexe:

Comment compiler les différents exercices ?

Pour ce projet un makefile à été créé, il ne vous reste donc qu'à taper la commande suivante: **make Chrono.t**

Une fois ceci un fichier Chrono.o à été créé ainsi qu'un fichier Chrono.

Il ne vous reste plus qu'à taper la commande: **./Chrono** et le chrono va s'afficher dans votre terminal.

Attention ce dernier ne marche que pour le chrono finale pour compiler les autres projet il faut taper les commande suivantes:

- pour l'exercice 1: **gcc -ansi -pedantic ChronoSimple 1.c -Wall**
- pour l'exercice 2: **gcc -ansi -pedantic ChronoMoyen.c -Wall -Incurses**

En "bonus" pour le chrono final, ce dernier comporte un fichier Chrono.c, avec toutes l'implémentation des fonctions. Ainsi que Chrono.h ou toutes les descriptions, documentation des fonctions est structure de Chrono.C.

Et le main.c contenant le "corp" du programme.