

TME 1 : RAPPEL C

L'objectif de ce TME est de vous familiariser avec la manipulation de programmes C sous Unix.

Fonctions utiles au TME :

La commande shell :

`man`

La fonction C :

`atoi, rand, srand, getrusage`

1. ECRIRE UN PROGRAMME C AVEC DES ARGUMENTS

1.1

Ecrivez un programme permettant de calculer le maximum d'une liste d'entiers passés par la ligne de commande. Générez l'exécutable, testez votre programme

2. ECRIRE UN PROGRAMME C QUI MANIPULE UN TABLEAU

On souhaite manipuler un tableau de taille `NMAX`.

La taille du tableau `NMAX` et les prototypes des fonctions doivent être définies dans le fichier `tab.h`, Les fonctions de manipulation du tableau sont dans le fichier `tab.c` et le programme principal (le « main ») sera défini dans le fichier `maintab.c`.

2.1

Ecrivez une fonction `void InitTab(int *tab, int size)` qui initialise le tableau `tab` de taille `size`, précédemment allouée. Chaque élément du tableau contient une valeur aléatoire entre 0 et 9. Pour générer aléatoirement les valeurs, vous utiliserez les fonctions `srand` et `rand`

2.2

Ecrivez une fonction `void PrintTab(int *tab, int size)` qui affiche les valeurs du tableau

2.3

Ecrivez une fonction `main` qui appelle `InitTab` et `PrintTab` pour un tableau initialisé sur la pile (en tant que variable locale du `main`) et dans le tas (en utilisant `malloc`).

2.4

Ecrivez un makefile pour générer l'exécutable et testez votre programme pour NMAX =10.

2.5

Ecrivez une fonction `int SumTab(int *tab, int size)` qui retourne la somme des éléments du tableau

2.6

Ecrivez une fonction `int MinSumTab(int *min, int *tab, int size)` qui retourne la somme des éléments du tableau et met dans la variable `min` l'élément minimum du tableau

2.7

Testez vos 2 fonctions dans le main.

On veut dans la suite observer la taille mémoire du programme.

2.8

Dans le fichier `maintab.c` écrivez la fonction `void PrintMem()` qui affiche la mémoire résidente occupée par votre programme. Cette fonction utilisera l'appel système `getrusage` pour afficher le champ `ru_maxrss` de la structure `rusage`.

2.9

Pour un tableau de grand taille (NMAX = 1000000), appelez `PrintMem()` avant et après `InitTab`. Que constatez-vous ? A quel moment la mémoire est effectivement allouée ?