L2 Programmation Impérative
TD1. Fichiers

1 Crible Erasthostene

L'algorithme du crible d'Erasthostene est une technique pour trouver tous les nombres premiers compris entre 1 et un entier n :

1. Créer un tableau contenant tous les entiers de 2 à n.
2. Prendre le plus petit élément non barré dans le tableau.
3. Cet élément est un premier.
4. Barrer cet élément et tous ses multiples.
5. Répéter à partir de l'étape 2 jusqu'à ce qu'il n'y ait plus d'élément non barré.

Ecrire un programme qui utilise cet algorithme pour écrire les nombres premiers de 1 à 100 dans le fichier */home/user/TD1/premier.txt*

2 Fichiers texte

1. Ecrire une fonction pour numéroter les lignes d'un fichier (commande nl sous UNIX). La fonction a deux paramètres d'entrée qui sont les noms (chemins d'accès) des fichiers, le premier indique un fichier déjà existant et le deuxième sera créé en ajoutant les numéros des lignes au premier.
2. On veut afficher un fichier par colonne (commande cut sous Unix). Ecrire une fonction qui prend 3 paramètres d'entrée, F1 : le nom du premier fichier, n : un entier, F2 : le nom du deuxime fichier. On copie le n-ième colonne du F1 sur F2 (qui sera créé).

3 Nombre en base B

On appelle *chiffre en base B* tout nombre entier compris entre 0 et $B-1$. Donc, une fois la base fixée, il y a un nombre fini de chiffres, et en théorie, il est toujours possible de représenter chaque chiffre par un symbole.

Propriété 1 *Soit $B > 1$ un entier. Soit $n > 0$ un entier. Alors il existe une unique suite d'entiers naturels $(a_k, a_{k-1}, \dots, a_1, a_0)$ telle que :*
- $a_k \neq 0$
- *Pour tout i, a_i est un chiffre en base B.*
- $n = a_0B^0 + a_1B^1 + a_2B^2 + \cdots + a_kB^k$.

Cette suite s'appelle le *développement en base B* de n. Les a_i s'appellent les *chiffres* de n. On écrit $n = a_ka_{k-1}\dots a_1a_0$. Dans l'usage courant en informatique, les bases les plus utilisées sont la base 10 et la base 2. Un entier écrit en base 2 est dit sous forme *binaire*, en base 10 sous forme *décimale*. On appelle *bits* les chiffres d'un entier en base 2 ('binary digit' en anglais). Les bases 8 et 16 sont aussi utilisées par les informaticiens : on parle de nombres donnés en *octal* et en *hexadécimal*.

Changement de base : Pour trouver les chiffres d'un nombre n dans une base B, il suffit d'être capable de calculer quelques divisions Euclidiennes :

$q_0 = n$
Pour tout $i \geq 0$ on définit a_i et q_{i+1} par :
$q_i = q_{i+1}B + a_i,$ avec $0 \leq a_i < b$

On vérifie qu'à partir d'un rang $k+1$, la suite (a_i) devient nulle. Et on a, en base B : $n = a_k a_{k-1} \ldots a_0$.

On représente un entier en base B par un tableau d'entiers contenant ses chiffres. Pour marquer la fin du nombre, on utilise B (qui n'est pas un chiffre en base B). Par exemple pour représenter 6234 en base 10, les cases du tableau sont les suivantes : 6, 2, 3, 4, 10. Le 10 est simplement là pour indiquer à l'ordinateur qu'il faut s'arrêter de lire. Pour représenter 31 en base 2, les cases du tableau contiennent : 1, 1, 1, 1, 1, 2. Pour représenter 16 en base 16 : 1, 0, 16. Pour représenter 31 en base 16 : 1, 15, 16.

1. Ecrire une fonction qui prend en entrée deux entiers n, B et un nom de tableau d'entiers T. La fonction doit instancier T avec les chiffres de n en base B.
2. Ecrire la fonction réciproque de la précédente : on donne B et T, et la fonction retourne l'entier correspondant.
3. Ecrire une fonction permettant de convertir un nombre d'une base à une autre. Les deux fonctions des questions précédentes peuvent être utiles.
4. Ecrire une fonction qui prend en entrée trois noms de tableau T_1, T_2, T_3, et qui instancie T_3 avec la représentation de la somme des entiers représentés par T_1 et T_2. Cette fonction retourne un entier qui est la taille de T_3.

4 Schéma de Horner pour évaluer un polynôme

Le polynôme $P(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots a_1 x^1 + a_0 x^0$ est défini par le tableau de coefficients A (nous supposons que le degré n est limité à 10).

1. Ecrire une fonction pour saisir le tableau de coefficients A de degré n. Les paramètres d'entrée de cette fonction sont un entier (n) et un tableau de réel (A).
2. Ecrire une fonction pour évaluer le polynôme pour un x donné. Les paramètres d'entrée de cette fonction sont un entier (n), un tableau de réel (A) et un réel (x). La fonction retourne une valeur réelle $(P(x))$.
3. Ecrire une fonction pour évaluer le polynôme avec le schéma de Horner. $P(x) = (\cdots(((a_n)x + a_{n-1})x + a_{n-2})x + \cdots + a_1)x + a_0$. Les paramètres d'entrée de cette fonction sont un entier (n), un tableau de réel (A) et un réel (x). La fonction retourne une valeur réelle $(P(x))$.
4. Comptez et comparez les nombres de multiplication effectuées dans 2 et 3.
5. Ecrire une fonction pour trouver la somme de deux polynômes. La fonction a 5 paramètres : n1 : le degré du premier polynôme, A1 : le tableau de coefficients du premier polynôme, n2 : le degré du deuxième polynôme, A2 : le tableau de coefficients du deuxième polynôme, A3 : le tableau de coefficients du polynôme qui est la somme de deux polynômes. La fonction retourne un entier qui est le degré du polynôme A3.