INF 111, L1, 2004/2005

CC - Quick 1

Novembre 2004

Interrogation "Quick"

Laure Gonnord
`http://www-verimag.imag.fr/~gonnord/`

Aucun document autorisé - Durée, 1h
On demande d'EXPLIQUER les programmes écrits

Exercice 1 – 2pts

Écrire une fonction `est_symetrique` qui prend un argument un tableau d'entiers à 2 dimensions `N/N`, et qui décide si ce tableau est symétrique. Un tableau t est symétrique si il vérifie :

$$\forall i, j, t[i][j] = t[j][i]$$

Votre fonction doit renvoyer 1 si le tableau est symétrique, 0 sinon.

Exercice 2 – 2pts

Que fait le programme suivant :

```
int main (void)
{
  int i, tab[5] ;
  for (i=0 ; i<=4 ; i=i+1)
    {tab[i] = i ;}

  for (i=0 ; i<=4 ; i=i+1)
    {
      tab[i] = tab[4-i] ;
      printf ("tab[%i]=%i\n",i,tab[i]) ;
    }
  return (0) ;
}
```

Problème – 6pts

Pour la suite du problème, on fixe une constante `N` égale à 500.

Le but de ce problème est de trouver tous les nombres premiers inférieurs à `N`. Pour cela, on va remplir un tableau `premier` de la façon suivante : `premier[i]` vaut 1 si et seulement si i est un nombre premier. Pour cela, on utilise la propriété suivante :

Un nombre est premier si et seulement si il n'est pas multiple d'un nombre premier plus petit que lui.

Le principe de l'algorithme est le suivant :
- Au début on ne sait rien, on considère donc que tous les nombres sont premiers (sauf 0 et 1 qui ne comptent pas).

- Le nombre 2 est premier, on sait alors que tous ses multiples ne sont pas premiers, et on peut les éliminer (mettre les cases correspondantes du tableau `premier` à 0).
- Le nombre 3 n'a pas été éliminé, il est donc premier et on peut éliminer tous ses multiples.
- Le nombre suivant qui n'a pas été éliminé est 5, on peut alors éliminer tous ses multiples.
- On recommence ainsi jusqu'à ce qu'on ait plus aucun nombre à éliminer.

Question 1. (1pt) Écrire une procédure `initialise` qui prépare le tableau tel qu'il doit être au début de l'algorithme. La signature de cette procédure doit être :

```
void initialise(int premier[N])
```

Question 2. (1,5pt) Écrire une procédure `elimine_multiples` qui prend en argument le tableau `premier` et un entier i, et qui élimine tous les multiples de i. Elle doit avoir la signature suivante :

```
void elimine_multiples(int premier[N], int i)
```

Question 3. (2pts) Écrire une procédure `crible` qui prend en argument le tableau `premier` déjà initialisé et qui effectue l'algorithme de crible décrit au-dessus. À la fin de cette procédure, le tableau doit vérifier :

`premier[i]` $= 1$ si et seulement si i est premier

Question 4. (1,5pts) Écrire un programme qui affiche tous les nombres premiers plus petits que la constante `N`, un nombre premier par ligne.

Question Bonus

Expliquer rapidement comment concaténer deux chaînes de caractères représentées par les tableaux `u` et `v`. On ne demande pas de code C.

Sources

- Le problème sur le crible d'Érathostène a été largement pompé sur le quick de Lionel Eyraud 2003/2004.
- Idem pour la question sur les tableaux symétriques.
- Le quick de Sylvain Daudé, 2003/2004 pour le programme mystère.