

Développement initiatique

Sujet 5 : divers exercices

1 Tables de multiplication

Ecrire une procédure `multiplication (n : entier) retourne void` qui permet d'afficher à l'écran la table de multiplication du nombre n par les entiers de 1 à 10, en respectant la forme suivante (pour $n = 4$) :

```
1 * 4 = 4
2 * 4 = 8
3 * 4 = 12
4 * 4 = 16
5 * 4 = 20
6 * 4 = 24
7 * 4 = 28
8 * 4 = 32
9 * 4 = 36
10 * 4 = 40
```

2 Chiffres romains

Rappelons d'abord la valeur des chiffres romains.

Chiffre romain	Nombre équivalent en chiffres arabes (base 10)
M	1000
D	500
C	100
L	50
X	10
V	5
I	1

Les nombres en chiffres romains sont formés selon les règles suivantes :

La traduction d'un entier compris entre 0 et 3999 de sa représentation usuelle en base 10 à sa représentation en chiffres romains se fait chiffre par chiffre et dans le même ordre (unités à droite).

Il n'y a pas de chiffre romain représentant le nombre 0.

Le chiffre des unités (de 0 à 9) se traduit respectivement par (rien), I, II, III, IV, V, VI, VII, VIII,

IX (noter que le préfixe I devant V et X a un rôle soustractif et non additif).

De même, le chiffre des dizaines (de 0 à 9) se traduit respectivement par (rien), X, XX, XXX, XL, L, LX, LXX, LXXX, XC.

De même, le chiffre des centaines (de 0 à 9) se traduit respectivement par ... Le chiffre des milliers (de 0 à 3) se traduit respectivement par ...

Exemples :

- 1999 se traduit par MCMXCIX
- 334 se traduit par CCCXXXIV
- 487 se traduit par CDLXXXVII

1. Ecrire la procédure suivante :

```
fonction chiffreUnites (nb : entier) retourne void
// Pré-requis : 0 <= nb <= 9
// Action : affiche à l'écran la représentation de nb en chiffres
romains.
```

2. Ecrire la procédure suivante :

```
fonction chiffreDizaines (nb : entier) retourne void
// Pré-requis : 0 <= nb <= 9
// Action : affiche à l'écran la représentation en chiffres romains de nb*10.
```

3. Ecrire les procédures `chiffreCentaines` et `chiffreMilliers` définies de façon analogue aux procédures précédentes.

4. Ecrire la procédure suivante :

```
fonction afficherEnChiffresRomain (nb : entier) retourne void
// Pré-requis : 0 <= nb <= 3999
// Action : affiche à l'écran la représentation de nb en chiffres romains.
```

5. Ecrire la procédure principale `chiffresRomains` qui saisit au clavier un entier compris entre 0 et 3999 et affiche à l'écran sa représentation en chiffres romains.

3 Carré parfait

Ecrire les fonctions suivantes en utilisant si possible les fonctions définies dans les questions précédentes.

1. Ecrire une fonction `racineParfaite` qui retourne la racine carrée entière n d'un nombre entier c donné, si c est un carré parfait, c'est-à-dire si $c = n \times n$. Sinon la fonction retourne -1 .
2. Ecrire une fonction `estCarreParfait` qui détermine si un entier donné est un carré parfait.
3. Ecrire une fonction qui détermine si deux entiers donnés peuvent être les côtés de l'angle droit d'un triangle rectangle dont les 3 côtés sont des nombres entiers.

4 Nombres amis

1. Deux nombres p et q sont dits *amis* si p vaut la somme des diviseurs de q (autres que q) et q vaut la somme des diviseurs de p (autres que p). Par exemple 220 et 284.
Ecrire une fonction qui détermine si deux nombres entiers p et q sont amis.
2. Ecrire une procédure qui permet d'afficher à l'écran les couples de nombres amis parmi les nombres inférieurs ou égaux à un entier donné.

5 Compactage par suppression des valeurs non significatives

On suppose que dans un vecteur d'entiers donné seules les valeurs non nulles sont significatives. Pour pouvoir traiter plus rapidement ce vecteur, il est préférable de regrouper les valeurs significatives. Il est donc demandé de déplacer les valeurs non nulles, sans en changer l'ordre, dans les éléments de faible indice du vecteur.

Ecrire une fonction `compactage` qui renvoie un tableau compacté du tableau non vide donné en paramètre.

Exemple :

Vecteur initial : 4 0 5 2 0 0 2 4 0 4

Vecteur résultat : 4 5 2 2 4 4 0 0 0 0

Variante : Ecrire une fonction qui modifie le tableau donné en paramètre. (Demander à l'enseignant une explication sur le passage de paramètres par variable/pointeur/référence/donnée-résultat.)

6 Compactage par suppression des répétitions

On suppose que dans un vecteur donné seules les valeurs non nulles sont significatives. On souhaite supprimer les valeurs non nulles en double. Pour cela, on remplacera les valeurs significatives en double par des valeurs non significatives et on déplacera les valeurs significatives dans les éléments de faible indice du vecteur. Ecrire une fonction qui résolve ce problème.

Exemple :

Vecteur en paramètre : 4 0 5 2 0 0 2 4 0 4

Vecteur résultat : 4 5 2 0 0 0 0 0 0 0

7 Recherche du plus long plateau dans un vecteur

Il s'agit de trouver la plus longue suite d'éléments consécutifs égaux dans un vecteur de caractères donné. Ecrire une procédure qui affiche l'indice de début et la longueur de ce plus long plateau.

Exemple, avec le vecteur donné : A Z E R R R Z E Z Z

Le plus long plateau commence à l'indice 3 (élément 'R') et sa longueur vaut 3.

8 Nombre le plus fréquent

Ecrire une fonction qui retourne le nombre le plus fréquent dans un tableau (non vide) d'entiers naturels donné.

On pourra utiliser une fonction auxiliaire qui modifie le tableau en paramètre ¹.

1. C'est ce que l'on appelle le passage de paramètres par données-résultats (ou référence)), par défaut pour les tableaux dans de nombreux langages et étudié dans quelques semaines...