

CONTRÔLE MAPLE DU 13 NOVEMBRE 2009

CPES FEYDER II

Exercice 1 : Écrire une procédure **somprem** qui prend deux entiers m et n en arguments et qui calcule la somme des entiers premiers entre m et n (on pourra supposer que $m \leq n$).

Exercice 2 :

- (1) Écrire un programme **somdiv** qui calcule la somme des diviseurs de l'entier n (1 et n compris) ;
- (2) Écrire ensuite un programme **parfait** qui répond **true** si n est un nombre parfait (c'est-à-dire égal à la moitié de la somme de ses diviseurs) et **false** dans le cas contraire. Par exemple, 6 est parfait puisqu'il a pour diviseurs 1, 2, 3, 6 et leur somme est 12 ;
- (3) Écrire un programme **listeparfaits** qui prend un entier n en argument et renvoie la liste de tous les nombres parfaits inférieurs ou égaux à n .

Exercice 3 : Un nombre **plouton** est un entier qui a strictement plus de diviseurs que ses prédécesseurs (les entiers strictement inférieurs à ce nombre). Par exemple, 4 est un nombre plouton puisqu'il admet trois diviseurs 1, 2, 4, alors que 3 et 2 n'ont que deux diviseurs et 1 un unique.

- (1) Écrire une procédure **nbdiv** qui calcule le nombre de diviseurs d'un entier n ;
- (2) Écrire une procédure **plouton** qui permet d'afficher la liste des entiers ploutons inférieurs à N (l'entier N est passé en argument de la procédure).

Exercice 4 : Un couple d'entiers (a, b) est **amiable** si, et seulement si, a est égal à la somme des diviseurs propres¹ de b , b est la somme des diviseurs propres de a , et enfin $a \neq b$.

Date: 13-11-2009.

1. Un diviseur **propre** d'un entier n est un diviseur de n qui est distinct de n .

Écrire une procédure Maple qui prend un entier N en argument et retourne la liste des listes $[a, b]$ où $a < b \leq N$ et (a, b) est amiable. Par exemple : `amiable(2000) = [[220, 284], [1184, 1210]]`.

Exercice 5 : Il est possible de représenter chaque entier naturel par un ensemble de la manière suivante. Soit E un ensemble. On définit le **successeur** E^+ de E par $E^+ = E \cup \{E\}$. La version ensembliste des entiers est donc

$$\begin{aligned} 0 &= \emptyset \\ 1 &= 0^+ (= \{0\}) = \{\emptyset\} \\ 2 &= 1^+ (= \{0, 1\}) = \{\emptyset, \{\emptyset\}\} \\ &\dots \\ 9 &= 8^+ (= \{0, 1, 2, 3, 4, 5, 6, 7, 8\}) \\ &\dots \end{aligned}$$

En Maple, l'ensemble vide est `{}` et l'expression « `{a}` » désigne le singleton a . L'union est quant à elle réalisée par l'opérateur `union`.

Écrire une procédure `entier` qui code un certain entier n sous sa forme ensembliste dans laquelle n'apparaissent que des accolades ouvrantes “`{`” et fermantes “`}`” (écrire une version récursive et une autre à l'aide d'une boucle `for` ou `while`).

Exercice 6 : On dit qu'un entier n est un **palindrome de base dix** si, et seulement si, son écriture en base dix, $(c_N c_{N-1} \dots c_1 c_0)_{dix}$ (donc

$$n = \sum_{i=0}^N c_i 10^i) \text{ vérifie}$$

$$c_i = c_{N-i}$$

pour tout i , $0 \leq i \leq N$.

Écrire une procédure `palindrome` qui détermine si un nombre donné est un palindrome de base dix.