

Concours Informatique Luxembourgeois 2001

Epreuve de Sélection Préliminaire

Problème I (Mini) - Nombres premiers consécutifs

Un nombre premier est un entier positif qui admet exactement 2 diviseurs, à savoir 1 et l'entier lui-même. Deux nombres premiers sont appelés consécutifs si leur différence est soit 1 soit 2.

Exemples

17 et 19 sont deux nombres premiers consécutifs
 2 et 3 sont deux nombres premiers consécutifs
 67 et 71 ne sont pas deux nombres premiers consécutifs

Ecrivez un programme qui admet en entrée les deux bornes délimitant un intervalle entier quelconque (ces deux valeurs sont saisies au clavier) et qui affiche à l'écran tous les couples de nombres premiers consécutifs contenus dans cet intervalle à raison d'un couple par ligne.

Exemples

Entrée: 10 50	Sortie: 11 13 17 19 29 31 41 43	Entrée: 30 50 Sortie: -
----------------------	---	---------------------------------------



Sauvegardez le programme sur la disquette sous le nom PREMIERS.xxx (avec xxx=PAS ou XXX=C). La disquette doit également contenir le fichier binaire exécutable PREMIERS.EXE correspondant au programme.

Problème II (Midi) - Représentation de dates

Considérons des nombres entiers à trois chiffres de 111 à 999. Certains de ces nombres peuvent représenter une date contenant le jour et le mois. (Admettons que le février ne comporte que 28 jours.)

Exemples

123 représente le 12 mars	319 ne représente aucune date
307 représente le 30 juillet ou bien le 2 juillet	913 ne représente aucune date

Ecrivez un programme qui produit le fichier texte DATES.OUT qui contient la liste de tous les nombres à 3 chiffres compris entre 111 et 999 qui représentent une ou plusieurs dates avec leur(s) date(s) respectif/ves.

Exemple

Fichier de sortie:	111 11 janvier ou 1 novembre
	112 11 février ou 1 décembre
	113 11 mars
	
	999 pas de date(s) reconnue(s)



Sauvegardez le programme sur la disquette sous le nom DATES.xxx (avec xxx=PAS ou XXX=C). La disquette doit également contenir le fichier binaire exécutable DATES.EXE correspondant au programme.

Problème I I a (Maxi) - Système de numération romain

Le système de numération romain repose sur un ensemble de 7 symboles (appelés chiffres romains) dont voici la table des équivalences avec le système décimal bien connu et les principes de la conversion d'un système numérique dans l'autre.

Equivalences: $1 \equiv I$ $5 \equiv V$ $10 \equiv X$ $50 \equiv L$ $100 \equiv C$ $500 \equiv D$ $1000 \equiv M$

Principe de la conversion du système décimal en système romain

$$888 = 800 + 80 + 8 \rightarrow DCCC + LXXX + VIII = DCCCLXXXVIII$$

$$1604 = 1000 + 600 + 0 + 4 \rightarrow M + DC + \text{rien} + IV = MDCIV$$

Principe de la conversion du système romain en système décimal

$$DCCXLVII = D + CC + XL + V + II \rightarrow 500 + 200 + 40 + 5 + 2 = 747$$

$$CMXCIX = CM + XC + IX \rightarrow 900 + 90 + 9 = 999$$

Ecrivez un programme qui effectue la conversion du système décimal en système romain et inversement. Le programme lit du clavier un nombre dans un système numérique donné et convertit ce dernier dans l'autre système – suivant le choix de l'utilisateur (prendre 'D' pour la conversion romain > décimal et 'R' pour la conversion décimal > romain).

Exemples

Entrée: MMMCMXCIX

D

Sortie: 3999

Entrée: 1997

R

Sortie: MCMXCVII



Sauvegardez le programme sur la disquette sous le nom DEC2ROM.xxx (avec xxx=PAS ou XXX=C). La disquette doit également contenir le fichier binaire exécutable DEC2ROM.EXE correspondant au programme.

Problème I I b (Maxi) - Carrés magiques

Considérons un tableau avec N colonnes et N lignes avec N=3 resp. 5. Pour que ce tableau soit un carré magique de degré N, il faut mettre des nombres différents dans les N×N cases, de façon à ce que les règles suivantes soient vérifiées:

$$\Sigma \text{ des nombres des lignes} = \Sigma \text{ des nombres des colonnes} = \Sigma \text{ des nombres des diagonales}$$

($\Sigma \equiv$ somme)

Pour un carré magique de degré N, cette somme est égale au nombre du milieu multiplié par N. Il faut en plus qu'aucun nombre ne figure deux fois dans les cases.

Exemple: carré magique de degré 3

4	9	2
3	5	7
8	1	6

Ecrivez un programme qui construit un carré magique de degré N (avec N=3 ou N=5). Le programme admet comme entrée un fichier texte MAGIC.IN qui comporte 3 nombres séparés par un espace, dont le premier correspond au degré du carré magique, le deuxième au nombre dans la première case (en haut à gauche) et le troisième au nombre de la case centrale. Le fichier texte de sortie MAGIC.OUT correspond au carré magique.

Exemple: carré magique de degré 3

Fichier d'entrée:

3	5	7,5
---	---	-----

Fichier de sortie:

5	14	3,5
6	7,5	9
11,5	1	10



Sauvegardez le programme sur la disquette sous le nom MAGIC.xxx (avec xxx=PAS ou XXX=C). La disquette doit également contenir le fichier binaire exécutable MAGIC.EXE correspondant au programme.