



Concours Informatique Luxembourgeois 2006

Epreuve de Sélection Préliminaire



Cotation sur 100 points

Problème I - Paires amicales

20 points

Deux nombres entiers positifs sont des paires amicales, si la somme des diviseurs du premier est égale au deuxième et si la somme des diviseurs du deuxième est égale au premier. La somme des diviseurs se comprend sans le nombre lui-même.

Exemple

220 et 284 sont des paires amicales, car

- les diviseurs de 220 sont: 1, 2, 4, 5, 10, 11, 20, 22, 44, 55, 110, 220

- la somme des diviseurs de 220 est égale à: $1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110 = 284$

et

- les diviseurs de 284 sont: 1, 2, 4, 71, 142, 284

- la somme des diviseurs de 284 est égale à: $1 + 2 + 4 + 71 + 142 = 220$

Problème

Ecrivez un programme qui recherche toutes les paires amicales entre 1 et $2^{15}-1$ (32767) et qui affiche les nombres trouvés à l'écran.



Remettez le programme sous le nom AMICALES.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable AMICALES.EXE correspondant au programme.

Problème II - Occurrences

20 points

Problème

Ecrivez un programme qui compte toutes les occurrences de lettres dans un texte donné. Les lettres sont à convertir en majuscules. Les caractères qui ne représentent pas de lettres (espace, signes de ponctuation etc.) sont à ignorer.

Entrée

Le programme accepte comme entrée le fichier texte OCC_IN.TXT qui contient le texte à analyser sur une seule ligne.

Sortie

Le programme affiche ses résultats à l'écran.

Exemple d'entrée et de sortie

OCC_IN.TXT: Alle meine Entchen schwimmen auf dem See.

Ecran:

A	2
B	0
C	2
D	1
E	9
.	
.	
.	
Y	0
Z	0



Remettez le programme sous le nom OCC.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable OCC.EXE correspondant au programme.

Problème III - Sudoku

30 points

Sudoku est un jeu de réflexion à la mode. On trouve toutes les informations nécessaires sur Internet au site www.sudoku.com

Problème

Chez Sudoku, chaque ligne, chaque colonne et chaque sous-carré 3×3 d'un tableau 9×9 contient obligatoirement tous les chiffres de 1 à 9. Si une ligne, une colonne ou un sous-carré contient déjà 8 chiffres on peut placer le neuvième chiffre manquant à la place vide de cette ligne, colonne ou sous-carré. Ecrivez un programme qui recherche toutes les lignes, toutes les colonnes et tous les sous-carrés contenant déjà 8 chiffres et qui ajoute les neuvièmes chiffres manquants à la bonne place.

Exemple

3				8		5		
				6		2		
8	2	6	1	5		3	9	7
6			5	4				2
	4		8	9	2			1
			2	3	6	1		8
							4	9
1	9		4	2	3	8	6	5
				9				7

Dans cet exemple, votre programme doit trouver:

1. que la troisième ligne contient huit chiffres et que le 4 y manque dans la sixième colonne
2. que la huitième ligne contient huit chiffres et que le 7 y manque dans la troisième colonne
3. que la huitième colonne contient huit chiffres et que le 3 y manque dans la quatrième ligne
4. que le cinquième sous-carré contient huit chiffres et que le 7 y manque dans la quatrième ligne et sixième colonne
5. enfin que la sixième colonne contient maintenant (!) huit chiffres et que le 5 y manque dans la septième ligne

Entrée

La configuration à analyser se trouve dans le fichier texte SUDO_IN.TXT. Ce fichier contient 9 lignes avec 9 chiffres séparés par un espace. Les cases vides sont définies par un zéro.

Exemple correspondant à l'exemple ci-dessus:

```
0 3 0 0 0 8 0 5 0
0 0 0 0 0 6 0 2 0
8 2 6 1 5 0 3 9 7
6 0 0 5 4 0 0 0 2
0 4 0 8 9 2 0 1 0
0 0 2 3 6 1 0 8 0
0 0 0 0 0 0 0 4 9
1 9 0 4 2 3 8 6 5
0 0 0 0 0 9 0 7 0
```

Sortie

Le résultat de votre programme est écrit dans le fichier texte SUDO_OUT.TXT

Exemple correspondant à l'exemple ci-dessus:

```
0 3 0 0 0 8 0 5 0
0 0 0 0 0 6 0 2 0
8 2 6 1 5 4 3 9 7
6 0 0 5 4 7 0 3 2
0 4 0 8 9 2 0 1 0
0 0 2 3 6 1 0 8 0
0 0 0 0 0 5 0 4 9
1 9 7 4 2 3 8 6 5
0 0 0 0 0 9 0 7 0
```

Les chiffres encadrés doivent être ajoutés par votre programme.



Remettez le programme sous le nom SUDOKU.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable SUDOKU.EXE correspondant au programme.

Problème IV - La maison de Saint Nicolas

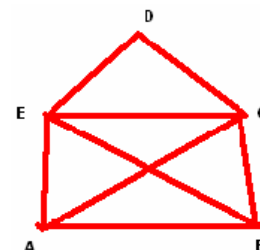
30 points

Soit la maison de Saint Nicolas, définie par les 5 points A, B, C, D et E, comme indiqué sur la figure ci-dessous. On peut trouver plusieurs solutions afin de dessiner la maison de Saint Nicolas sans soulever le crayon, donc en une seule ligne et sans passer par une ligne déjà existante, en partant d'un des 5 points.

Exemple

Quelques solutions possibles sont représentées ci-dessous:

```
ABECDEACB
BACEDCBEA
BEDCAECBA
```



Problème

Ecrivez un programme qui recherche toutes les solutions pour dessiner la maison de Saint Nicolas et qui les écrit ligne par ligne dans le fichier texte NICO_OUT.TXT dans le format utilisé dans l'exemple.



Remettez le programme sous le nom NICOLAS.xxx, avec xxx=PAS ou C(PP). Remettez également le fichier binaire exécutable NICOLAS.EXE correspondant au programme.