



**Concours
canadien de
mathématiques**

Une activité du Centre d'éducation
en mathématiques et en informatique
Université de Waterloo, Waterloo (Ontario)

*Concours
canadien
d'informatique*

pour les bourses de  SYBASE®

Le mardi 29 février 2000

Problème J1

Calendrier

Vous écrivez un programme pour imprimer un calendrier pour un mois particulier donnant le jour où tombe le premier du mois de même que le nombre de jours dans le mois.

Votre programme doit prendre comme entrée un nombre entier qui représente le jour de la semaine où débute le mois (1 pour dimanche, 2 pour lundi, ..., 7 pour samedi), et un nombre entier qui représente le nombre de jours dans le mois (entre 28 et 31 inclusivement.) Vous pouvez supposer que toutes les données d'entrée seront valides.

L'entrée et la sortie ne se font pas des fichiers de/à pour cette question. Il faut donc s'attendre à une entrée sur clavier et à une sortie sur l'écran.

Exemple de séance. *L'entrée de l'utilisateur est en italiques*

Entrez le jour :

3

Entrez le nombre de jours dans le mois :

30

Dim	Lun	Mar	Mer	Jeu	Ven	Sam
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Problème J2

9966

Les chiffres 0, 1, et 8 se ressemblent étrangement si on les retourne à 180 degrés sur la page (image renversée). De même, le chiffre 6 ressemble beaucoup à un 9, et vice versa, quand il est renversé. Un nombre à plusieurs chiffres peut également se ressembler quand il est renversé; par exemple, 9966 et 10801, mais 999 et 1234 ne le peuvent pas.

Vous devez écrire un programme capable de compter combien de nombres dans un intervalle donné se ressemblent quand ils sont renversés sur la page. Par exemple, dans l'intervalle de 1...100, il y a 1, 8, 11, 69, 88 et 96.

Votre programme doit considérer comme intrant deux nombres entiers, m et n , qui définissent l'intervalle à vérifier, $1 \leq m \leq n \leq 32000$. La donnée de sortie de votre programme est la quantité de nombres susceptibles d'être renversés dans cet intervalle.

Vous pouvez supposer que toutes les sorties sont valides.

L'entrée et la sortie ne se font pas des fichiers de/à pour cette question. Il faut donc s'attendre à une entrée sur clavier et à une sortie sur l'écran.

Exemple de séance *L'entrée de l'utilisateur est en italiques.*

Entrez la limite inférieure de l'intervalle :

1

Entrez la limite supérieure de l'intervalle :

100

La quantité de nombres susceptibles d'être renversés est :

6

Note aux candidats : vous trouverez peut-être le problème S1J3, sur les appareils à sous, plus facile à solutionner que celui-ci, et vous voudrez peut-être vous y attaquer en premier.

Problème S1J3

Appareils à sous

Marthe emporte avec elle un pot rempli de pièces de 25 cents au casino dans l'espoir d'y faire fortune. Elle joue sur trois machines l'une après l'autre. Elle ne sait pas que les appareils sont tout à fait prévisibles. Chaque jeu coûte 25 cents. Le premier appareil paie 30 pièces de 25 cents chaque fois que le 35^e jeu est fait; le deuxième verse 60 pièces chaque fois que 100 jeux sont faits et le troisième verse 9 pièces à chaque 10 coups.

Votre programme doit entrer le nombre de pièces qu'a Marthe dans son pot (au moins une et moins de 1000) et le nombre de fois que chaque appareil a été actionné depuis la dernière fois qu'elle a payé.

Votre programme doit sortir le nombre de fois que Marthe peut jouer avant de ne plus avoir une seule pièce.

L'entrée et la sortie ne se font pas des fichiers de/à pour cette question. Il faut donc s'attendre à une entrée sur clavier et à une sortie sur l'écran.

Exemple de séance. *L'entrée de l'utilisateur est en italiques*

Combien de pièces Marthe a-t-elle dans son pot?

48

Combien de fois a-t-on joué sur le premier appareil depuis la dernière fois qu'il a payé?

3

Combien de fois a-t-on joué sur le deuxième appareil depuis la dernière fois qu'il a payé?

10

Combien de fois a-t-on joué sur le troisième appareil depuis la dernière fois qu'il a payé?

4

Marthe joue 66 fois avant de dépenser tout ce qu'elle a.

Problème S2J4

Les ruisseaux gazouilleurs

Des ruisseaux dévalent une montagne. Le flanc de la montagne est rocailleux et les ruisseaux se séparent et se rejoignent plus d'une fois. Au pied de la montagne, plusieurs ruisseaux forment des rivières. Votre tâche consiste à calculer le volume d'eau qui coule dans chacune des rivières.

À toute élévation donnée, se trouvent n ruisseaux étiquetés 1 à n de gauche à droite. En descendant la montagne, un des ruisseaux peut se diviser pour former une fourche gauche et une fourche droite, ce qui augmente le nombre de ruisseaux de 1, ou encore deux ruisseaux peuvent fusionner pour réduire de 1 le nombre total de ruisseaux. Après une séparation ou une fusion, les ruisseaux sont à nouveau numérotés consécutivement de gauche à droite. Il existe toujours au moins un ruisseau, mais jamais plus de 100.

La première ligne d'entrée contient n , le nombre initial de ruisseaux à une certaine altitude. Les lignes n suivantes indiquent le débit de chacun des ruisseaux de gauche à droite. Sur les flancs de la montagne, plusieurs séparations ou fusions se produisent. Pour chaque emplacement de **séparation**, il y a trois lignes d'entrée :

- une ligne contenant 99 (pour indiquer une séparation)
- une ligne contenant le numéro du ruisseau séparé
- une ligne contenant un numéro entre 0 et 100, le pourcentage du débit du ruisseau séparé qui coule vers la gauche. (Le reste coule vers la droite.)

Pour chaque emplacement de **fusion**, il y a deux lignes d'entrée :

- une ligne contenant 88 (pour indiquer une fusion)
- une ligne contenant le numéro du ruisseau qui fusionne avec le ruisseau à sa droite.

Le débit des deux ruisseaux fusionnés est combiné. Après la dernière séparation ou la dernière fusion, se trouve

- une ligne qui contient 77 (pour indiquer la fin de l'entrée)

Votre tâche consiste à déterminer le nombre de ruisseaux au pied de la montagne et le débit de chacun. Votre entrée est une séquence de nombres réels, arrondis au nombre entier le plus près, donnant le débit des ruisseaux de 1 à n .

Exemple d'entrée (fichier d'entrée : brooks.in)

```
3
10
20
30
99
1
50
88
3
88
2
77
```

Sortie pour l'exemple d'entrée (fichier de sortie: brooks.out)

Problème S3J5

Navigation

Chaque page web est identifiée par une chaîne de caractères connue sous le nom de URL (localisateur de ressources uniformes). Les pages web sont formatées à l'aide du HTML (langage hypertexte). Le HTML a plusieurs codes, collectivement connus comme markup, qui permettent à l'auteur de déterminer le format de la page et les liens avec d'autres pages. Pour le présent problème, nous ne nous intéressons qu'au markup qui détermine, dans une page donnée, les liens a avec d'autres pages.

Un lien dans une page porte la mention `` où *URL* est le URL d'une autre page. Un utilisateur qui voit une page contenant un lien peut cliquer sur ce lien pour voir une autre page.

Vous devez écrire un programme qui lit un numéro de pages web et leurs URL. Pour chaque lien de chacune des pages, vous devez imprimer le URL de la page qui contient le lien et le URL de la page à laquelle renvoie le lien.

Suite à la dernière page, vous aurez plusieurs paires de URL. Pour chacune de ces paires, vous supposerez que vous voyez la page déterminée par le premier URL, et préciserez s'il est possible de cliquer une séquence de liens de façon à voir la page identifiée par le deuxième URL. Le cas échéant, vous devrez imprimer « je peux naviguer *d'ici à là* » où *ici* et *là* sont les deux URL.

Sinon, vous devez imprimer « je ne peux naviguer *d'ici à là* ».

La première ligne de l'entrée contient un nombre entier $n \leq 100$, le nombre de pages web. Pour chaque page web, une ligne contient son URL, suivi de plusieurs lignes contenant la page. Le URL consiste en 80 caractères imprimables autres qu'un blanc, sans guillemets. La première ligne de la page sera `<HTML>` et la dernière sera `</HTML>`. Chaque page contiendra jusqu'à 100 liens dans le format décrit plus haut. Chaque lien sera contenu dans une ligne unique d'entrée. Les URL du lieu seront ceux des pages mentionnées dans l'entrée. Les mots clés du markup A, HREF, et HTML n'apparaîtront qu'en haut de casse. Après les n pages, se trouveront plusieurs paires de lignes donnant les URL requis par le problème, comme mentionné plus haut. La dernière ligne de l'entrée sera "The End". Pour chaque paire, imprimez le message approprié donné plus haut.

Exemple d'entrée (fichier d'entrée : surf.in)

```
3
http://ccc.uwaterloo.ca
<HTML> <TITLE>This is the CCC page</TITLE>
Hello there boys
and girls. <B>Let's</B> try <A HREF="http://abc.def/ghi"> a
little
problem </A>
</HTML>
http://abc.def/ghi
<HTML> Now is the <TITLE>time</TITLE> for all good people to program.
<A HREF="http://www.www.www.com">hello</A><A HREF="http://xxx">bye</A>
</HTML>
http://www.www.www.com
<HTML>
<TITLE>Weird and Wonderful World</TITLE>
</HTML>
http://ccc.uwaterloo.ca
http://www.www.www.com
http://www.www.www.com
http://ccc.uwaterloo.ca
The End
```

Sortie pour l'exemple d'entrée (fichier de sortie : surf.out)

Lien de <http://ccc.uwaterloo.ca> à <http://abc.def/ghi>

Lien de <http://abc.def/ghi> à <http://www.www.www.com>

Lien de <http://abc.def/ghi> à <http://xxx>

Je peux naviguer de <http://ccc.uwaterloo.ca> à <http://www.www.www.com>.

Je ne peux naviguer de <http://www.www.www.com> à <http://ccc.uwaterloo.ca>

Problème S4

Golf

Roberta le Robot joue une partie parfaite de golf. Quand elle frappe la balle, elle va toujours directement vers le trou sur le vert et frappe toujours exactement la distance précisée pour le bâton. Chacun des gestes est considéré comme un coup et le but du jeu de golf est de diriger la balle depuis le té jusqu'au trou en frappant le moins de coups possible. Roberta a besoin d'un programme pour choisir la meilleure combinaison de bâtons pour atteindre le trou en le moins de coups possible. Elle doit aussi décider si la tâche est impossible, dans quel cas elle acceptera la défaite sans mot dire. Roberta peut transporter jusqu'à 32 bâtons et la distance totale ne doit pas excéder 5280 mètres.

L'entrée

La première ligne de l'entrée donne la distance entre le té et le trou, un nombre entier de mètres entre 1 et 5280. La ligne suivante détermine le nombre de bâtons, entre 1 et 32. Pour chaque bâton, suit une ligne qui donne la distance, en mètres, sur laquelle le bâton frappera la balle, soit un nombre entier entre 1 et 100. Il n'y a pas deux bâtons qui frappent la même distance.

La sortie

Si Roberta peut amener la balle du té jusqu'au trou sans le dépasser, imprimez « Roberta gagne en n coups » où n est minimisé. Si Roberta ne peut frapper la balle du té jusqu'au trou, imprimez « Roberta reconnaît sa défaite ».

Exemple d'entrée (fichier d'entrée : golf.in)

```
100
3
33
66
1
```

Sortie pour l'exemple d'entrée (fichier de sortie : golf.out)

Roberta gagne en 3 coups.

Problème S5

Moutons et coyotes

Plusieurs moutons se trouvent sur un terrain carré de 1000 sur 1000. Un coyote fait son entrée sur le terrain à un point donné de sa limite sud et dévore le mouton le plus près de son entrée, choisissant arbitrairement sa victime si plus d'un mouton se trouve à égale distance. Le coyote, rassasié, quitte le terrain.

Vous devez déterminer quel mouton a été dévoré par le coyote.

Supposons que le coin sud-ouest est situé à (0.00, 0.00), le coin nord-ouest (0.00, 1000.00), le coin nord-est à (1000.00, 1000.00) et le coin sud-est à (1000.00, 0.00). La première ligne d'entrée donne le nombre de moutons, entre 1 et 100. Pour chaque mouton, suit une paire de lignes donnant ses coordonnées à l'intérieur du champ (entre 0.00 et 1000.00).

Pour chaque mouton susceptible d'être dévoré, imprimez une ligne « le mouton à (x, y) risque d'être dévoré », où x et y marque l'endroit où se trouve le mouton, à deux décimales près. Le mouton peut être désigné dans n'importe quel ordre dans le résultat.

Exemple d'entrée (fichier d'entrée : sheep.in)

```
6
100.00
100.00
200.00
150.00
140.00
200.00
100.00
300.00
300.00
300.00
300.00
100.00
```

Sortie pour l'exemple d'entrée (fichier de sortie : sheep.out)

```
Le mouton à (100.00, 100.00) peut être dévoré.
Le mouton à (300.00, 100.00) peut être dévoré .
```