

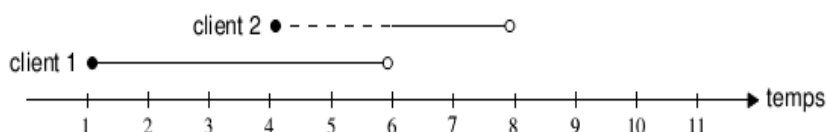
Supermarché

30 points

Vous êtes en charge d'étudier le comportement des clients d'un supermarché, au moment où ceux-ci passent à la caisse. En particulier, votre tâche consiste à calculer les instants de départ des clients (après paiement des marchandises), connaissant leurs instants d'arrivée à la caisse et le temps qu'a duré leur service (c'est-à-dire le temps durant lequel ils étaient pris en charge par la caissière).

Tâche

Soient N clients à une même caisse. Le fichier texte d'entrée contient les instants d'arrivée des clients ainsi que le temps passé à la caisse. Vous devez générer le fichier texte de sortie qui contient l'instant de départ de ces clients.



Prenons un exemple avec deux clients. Le premier arrive au temps 1 et son service dure 5 et le second arrive au temps 4 et son service dure 2. Le premier client va partir en 6 ($1 + 5$). Le second client va devoir attendre 2 que le premier client soit servi, puis encore 2 le temps d'être servi et il partira donc en 8.

Les N clients sont traités dans l'ordre dans lequel ils arrivent. Ils ne sont pas à réarranger pour optimiser leurs instants de départ.

Restrictions

- $1 \leq N \leq 1000$
- L'instant d'arrivée d'un client est un nombre entier strictement positif
- Le temps de service d'un client est un nombre entier strictement positif

Entrée et sortie du programme

Entrée

- La première ligne du fichier texte SUPERMARCHÉ.TXT contient un seul nombre entier positif: le nombre N de clients.
- Les N lignes suivantes contiennent deux nombres séparés par exactement un espace: le premier nombre désigne l'instant d'arrivée et le deuxième nombre désigne le temps de service de ce client. Ces N lignes sont triées par ordre croissant sur le temps d'arrivée!

Sortie

- Le fichier de sortie SUPERMARCHÉ_OUT.TXT doit contenir exactement N lignes qui contiennent chacune exactement un nombre: l'instant de départ du client respectif.

Exemples d'exécution

Exemple 1	Contenu du fichier <i>SUPERMARCHÉ.TXT</i> :	2
		1 5
		4 2
	Contenu du fichier <i>SUPERMARCHÉ_OUT.TXT</i> :	6
		8

Exemple 2	Contenu du fichier <i>SUPERMARCHE.TXT</i> :	6
		1 5
		2 2
		10 10
		12 7
		13 1
		29 4
	Contenu du fichier <i>SUPERMARCHE_OUT.TXT</i> :	6
		8
		20
		27
		28
		33

Remettez le programme sous le nom SUPERMARCHE.xxx, avec xxx=PAS ou C(PP) ou JAVA.

Concert

50 points

Un groupe de N amis se rend à un concert open-air de leur chanteuse favorite. Après avoir fait valider leurs tickets, ils inspectent attentivement le terrain rectangulaire du concert.

Pour que les fans du concert ne se dérangent pas entre-eux, l'organisation a décidé de subdiviser le terrain en "cellules". Chaque cellule peut contenir exactement un fan. Le terrain compte C colonnes et L lignes (il a donc en total $C*L$ cellules). Le groupe de N amis remarque que plusieurs cellules sont déjà occupées par des fans et décide de se placer de manière optimale dans les cellules restantes vides.



Tâche

Écrivez un programme qui lit à partir du fichier CONCERT.TXT la taille N du groupe ainsi que les cellules déjà occupées du terrain et qui détermine comment les N membres du groupe peuvent se placer de manière optimale dans les cellules. Le terrain rectangulaire compte C colonnes et L lignes qui forment $C*L$ cellules. Chaque cellule peut soit être vide ou être occupée par exactement un fan.

Deux membres du groupe sont voisins lorsqu'ils se trouvent dans deux cellules voisines. Deux cellules sont voisines lorsqu'elles ont un bord commun (horizontalement ou verticalement).

Le groupe veut se placer de telle manière à ce que chaque membre du groupe touche au moins un autre membre du groupe.

Toute solution qui respecte la règle ci-dessus sera acceptée.

Restrictions

- $1 \leq C \leq 100$ Le nombre de colonnes du champ
- $1 \leq L \leq 100$ Le nombre de lignes du champ
- $1 \leq N \leq C*L$ La taille du groupe

Entrée et sortie du programme

Entrée

- La première ligne du fichier texte CONCERT.TXT contient un seul nombre entier positif: la taille N du groupe.
- La deuxième ligne du fichier texte contient un seul nombre entier positif: le nombre de colonnes C du terrain.
- La troisième ligne du fichier texte contient un seul nombre entier positif: le nombre de lignes L du terrain.
- Les L lignes suivantes contiennent chacune exactement C lettres: la lettre 'X' symbolise que la cellule est occupée et la lettre 'O' symbolise que la cellule est vide. Toute autre lettre est interdite.

Sortie

Le programme doit générer le fichier de sortie CONCERT_OUT.TXT contenant la disposition du groupe sur le terrain. Il doit donc écrire dans le fichier la totalité du terrain et marquer les cellules où un membre du groupe est placé par le symbole '#'. Lorsqu'il n'est pas possible de placer les membres du groupe de la manière souhaitée, le terrain original inchangé doit être placé dans le fichier de sortie.

Exemples d'exécution

Exemple 1 Contenu du fichier *CONCERT.TXT*:

```

8
6
5
OOXOXO
OXXXOO
XXXOOX
OOXXOO
XOXXXO

```

Contenu du fichier *CONCERT_OUT.TXT*:

```

OOXOX#
OXXX##
XXX##X
OOXX##
XOXXX#

```

Exemple 2 Contenu du fichier *CONCERT.TXT*:

```

8
6
5
OOXOXO
OXXXOO
OXOOXX
OOXXOO
XOXXXO

```

Contenu du fichier *CONCERT_OUT.TXT*:

```

OOXOXO
OXXXOO
OXOOXX
OOXXOO
XOXXXO

```

Remettez le programme sous le nom CONCERT.xxx, avec xxx=PAS, C(PP) ou JAVA.

Code secret

20 points

Charlie trouve au fond du grenier de sa grand-mère un petit coffre-fort appartenant à son arrière-grand-père. Le problème est qu'il ne connaît pas la combinaison pour l'ouvrir. Il en parle à sa grand-mère qui se souvient que son père, amateur de mathématiques, avait choisi une méthode spéciale pour créer une combinaison tout en étant sûr de la retenir: il prenait une date, dans ce cas-ci, la date de son mariage et construisait à partir de ce nombre un nouveau nombre: plus spécifiquement le nombre le plus petit possible, tel que le produit des chiffres de ce nombre est égal au nombre de départ. Ses arrière-grands-parents s'étaient mariés le 15 mai 1928.

Tâche

Le but du programme est d'aider Charlie en écrivant un programme, qui lorsqu'on entre un nombre entier N , affiche le plus petit nombre entier R tel que le produit des chiffres constituant R est égal à N . Une fois ce programme réalisé, vous devez soumettre au programme les différentes formes de la date du mariage afin de trouver la combinaison ouvrant le coffre-fort.

Restrictions

- N est un entier tel que $1 \leq N \leq 1000000000$

Entrée et sortie du programme

Entrée

- Le programme lit le nombre N .

Sortie

- Si un nombre R existe tel que le produit de ses chiffres est égal à N , il affiche ce nombre R . Dans le cas contraire, il affiche le message « Pas de solution ».

Exemples d'exécution

Exemple 1 Si N est égal à 100, R vaut 455.

Affichage à la console: 100
455

Exemple 2 Si N est égal à 102, il n'y a pas de solution.

Affichage à la console: 102
Pas de solution

En sachant que son arrière-grand-père avait utilisé comme date de mariage la valeur 150528, remettez uniquement la combinaison trouvée par votre programme dans le fichier CODE.TXT.