

	Concours National de Programmation Togo 2014 « La bataille des cerveaux »	Campus Numérique Francophone 30 Mai 2014 (C, C++, JAVA)
---	---	---

## Problème A : Projet TAKA+ (Ballon bleu)

La filiale Logisoft du groupe MapCom chargée de l'intégration des systèmes et logiciels souhaite mettre à la disposition du gouvernement togolais, un outil de réduction des dépenses en carburant. Le projet est dénommé TAKA+.

Un des composants du projet TAKA+ est la détermination des itinéraires optimaux à utiliser par les véhicules administratifs.

Vous êtes chargé d'écrire un programme qui, étant donné une liste de villes et de distances séparant ces villes, donne la plus courte distance à parcourir pour aller d'une ville A à une ville B.

### Entrée Standard

La première ligne contient un seul entier **P**, ( $1 \leq P \leq 1000$ ), qui est le nombre de jeux de données. Chaque jeu de données commence par une ligne contenant le nombre **N** de lignes restantes dans le jeu de données ( $1 \leq N \leq 500$ ), suivi d'un espace, suivi du nom de la ville de départ A, suivi d'un espace, suivi du nom de la ville de destination B. Chacune de ces **N** lignes contient le nom d'une ville de départ suivi d'un espace, suivi du nom d'une ville d'arrivée, suivi de la distance (en kilomètres) entre les deux villes.

### Sortie Standard

Pour chaque jeu de données, vous devez générer une seule ligne de sortie contenant le nom de la ville A suivi d'un espace, suivi du nom de la ville B, suivi d'un espace, suivi de la plus courte distance à parcourir entre les villes A et B.

Exemple d'entrée	Exemple de sortie
1 4 Lome Dapaong Lome Kpalime 123 Dapaong Kara 218 Lome Kara 410 Kpalime Dapaong 563	Lome Dapaong 628

## Problem B: TOGO 2014

### (Ballon rouge)

Dans le cadre d'un projet d'optimisation de l'utilisation des encres par l'Administration, vous avez été recruté pour déterminer la fréquence d'apparition du nom "TOGO" dans les textes administratifs.

A partir d'une matrice  $n \times n$  de lettres majuscules, vous pouvez lire des mots du haut vers le bas ou de la gauche vers la droite. Voici l'exemple d'une matrice  $6 \times 6$ :

S	A	V	A	N	E
M	A	P	C	O	M
V	O	L	A	N	T
R	E	N	D	R	E
F	E	N	D	R	E
M	O	R	D	R	E

A partir de cette matrice, nous pouvons lire le mot **MAP** de la gauche vers la droite et le mot **ADD** du haut vers le bas.

En tant que grand programmeur, on vous demande de confirmer, étant donné une matrice, si le mot TOGO peut être lu de la gauche vers la droite ou du haut vers le bas ou ne peut pas être lu.

### Entrée standard

La première ligne contient un seul entier **P**, ( $1 \leq P \leq 1000$ ), qui est le nombre de jeux de données. Chaque jeu de données consiste en une ligne contenant la taille  $n$  ( $1 \leq n \leq 1000$ ) de la matrice, un espace et est suivie de  $n$  lignes contenant chacune  $n$  caractères.

### Sortie Standard

Pour chaque jeu de données, vous devez générer une seule ligne de sortie contenant le texte “TROUVE” if le mot TOGO a été trouvé dans la matrice (de la gauche vers la droite ou du haut vers le bas) ou “NON TROUVE” dans le cas contraire.

Exemple d'entrée	Exemple de sortie
3 6 SAVANE LEGUME VOLANT RENDRE FENDRE MORDRE 4 ABAC TOGO COLD MINE 7 CAMPUSF NATRESO APPLEME ACCOUNT ROUTERS AMIGOST MYSELFY	NON TROUVE TROUVE NON TROUVE

	<p><i>Concours National de Programmation Togo 2014 « La bataille des cerveaux »</i></p>	<p><i>Campus Numérique Francophone 30 Mai 2014 (C, C++, JAVA)</i></p>
---	---	---

## *Problème C : Nombres pairs (Ballon jaune)*

Un nombre pair est un nombre divisible par 2. Etant donné deux entiers  $A$  et  $B$  inférieurs à 1.000.000, vous devez trouver le nombre de nombres pairs strictement compris entre  $A$  et  $B$ .

### **Entrée Standard**

L'entrée commence par un seul entier  $N$ , ( $1 \leq N \leq 1000$ ), sur la première ligne, indiquant le nombre de cas qui vont suivre.

Les lignes restantes de l'entrée consistent en une série de paire d'entiers  $A$  et  $B$ , une paire d'entiers par ligne. Tous les entiers seront inférieurs à 1.000.000 et supérieurs à 0.

Vous devez traiter toutes les paires d'entier et pour chaque paire, déterminer le nombre de nombres pairs strictement compris entre  $A$  et  $B$ .

Vous pouvez supposer qu'aucune opération n'exigera l'utilisation d'un entier de plus de 32 bits.

### **Sortie Standard**

Pour chaque paire d'entiers en entrée  $A$  et  $B$ , vous devez sortir  $A$ ,  $B$  et le nombre de nombres pairs strictement compris entre  $A$  et  $B$ . Les trois nombres doivent être séparés par au moins un espace avec les trois nombres sur une même ligne et avec une ligne de sortie pour chaque ligne d'entrée. Les entiers  $A$  et  $B$  doivent apparaître dans la sortie dans le même ordre dans lequel ils apparaissent dans l'entrée et doivent être suivis par le nombre de nombres pairs strictement compris entre  $A$  et  $B$  (sur la même ligne).

	<i>Concours National de Programmation Togo 2014 « La bataille des cerveaux »</i>	<i>Campus Numérique Francophone 30 Mai 2014 (C, C++, JAVA)</i>
---	--	--

<b>Exemple d'entrée</b>	<b>Exemple de sortie</b>
4 1 10 100 200 201 210 900 1000	1 10 4 100 200 49 201 210 4 900 1000 49

## *Problème D : Plaque d'immatriculation (Ballon vert)*

Une plaque d'immatriculation togolaise est de la forme TG ABCD XY où A,B,C,D sont des chiffres compris entre 0 et 9 et X,Y des lettres majuscules comprises entre A et Z. Par exemple TG 2447 AL et TG 7070 AJ sont des numéros de plaques d'immatriculation togolaises.

Le Ministère des Travaux Publics et des Transports du Togo vous sollicite pour connaître le nombre de plaques d'immatriculation délivrées entre deux plaques d'immatriculation.

Votre rôle est d'écrire un programme pour répondre à cette requête.

Par exemple, entre les numéros TG 2447 AL et TG 2450 AL, il y a 4 plaques d'immatriculation délivrées.

### **Entrée Standard**

L'entrée commence par un seul entier  $N$ , ( $1 \leq N \leq 1000$ ), sur la première ligne, indiquant le nombre de cas qui vont suivre.

Les lignes restantes de l'entrée consistent en une série de paire de numéros de plaques d'immatriculation, une paire de numéros par ligne. Les éléments constitutifs des numéros sont séparés entre eux par un simple espace.

Vous devez traiter toutes les paires de numéros et pour chaque paire, déterminer le nombre de numéros de plaques d'immatriculation compris entre les deux numéros de la paire.

### **Sortie Standard**

Pour chaque paire de numéros en entrée, vous devez sortir le nombre de plaques d'immatriculation compris entre les deux numéros.

<b>Entrée standard</b>	<b>Sortie Standard</b>
3	4

	<p><i>Concours National de Programmation Togo 2014 « La bataille des cerveaux »</i></p>	<p><i>Campus Numérique Francophone 30 Mai 2014 (C, C++, JAVA)</i></p>
---	---	---

<p>TG2447 AL TG 2450 AL TG0001 AA TG 9999 AB TG1010 AC TG 1020 AC</p>	<p>19998 11</p>
---	---------------------

## *Problem E : Sequence Hailstone (Ballon blanc)*

La séquence hailstone est formée de la manière suivante:

- Si  $n$  est pair, divise le par 2 pour avoir  $n'$
- Si  $n$  est impair, multiplie le par 3 et ajoute 1 pour avoir  $n'$

Il est supposé que, pour n'importe quel nombre entier positif  $n$ , la séquence se terminera toujours dans un cycle répétitif 4, 2, 1, 4, 2, 1, ... Cela dit, quand  $n=1$ , nous dirons que la séquence est terminée.

Ecrire un programme pour déterminer la plus grande valeur dans la séquence pour un nombre entier  $n$  donné.

### Entrée Standard

La première ligne contient un seul entier  $P$ , ( $1 \leq P \leq 100000$ ), qui est le nombre de jeux de données. Chaque jeu de données doit être traité de façon identique et indépendante. Chaque jeu de données consiste en une seule ligne de l'entrée comportant deux entiers décimaux espacés. Le premier entier est le numéro du jeu d'essai. Le second entier est  $n$ , ( $1 \leq n \leq 100000$ ), qui est la valeur de départ.

### Sortie Standard

Pour chaque jeu de données, vous devez générer une seule ligne de sortie contenant le numéro du jeu de données, un seul espace, et la plus grande valeur dans la séquence commençant par et incluant  $n$ .

Exemple d'entrée	Exemple de sortie
4	1 1
1 1	2 16
2 3	3 101248
3 9999	4 100000
4 100000	



	Concours National de Programmation Togo 2014 « La bataille des cerveaux »	Campus Numérique Francophone 30 Mai 2014 (C, C++, JAVA)
---	---	---

## Problème F: Caractères répétitifs (Ballon orange)

Pour ce problème, vous allez écrire un programme qui prend une chaîne des caractères,  $S$ , et crée une nouvelle chaîne de caractères,  $T$ , avec chaque caractère se répétant  $R$  fois. C'est  $R$  copies du premier caractère de  $S$ , suivie par  $R$  copies du second caractère de  $S$ , et ainsi de suite. Les caractères valides de  $S$  sont les caractères alphanumériques suivants:

0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ\$%\*+-./:

### Entrée Standard

La première ligne contient un seul entier  $P$ , ( $1 \leq P \leq 1000$ ), qui est le nombre de jeux de données à suivre. Chaque jeu de données est une ligne unique de l'entrée constituée par le numéro de jeu de données  $N$ , suivi d'un espace, suivi par le nombre de répétitions  $R$ , ( $1 \leq R \leq 8$ ), suivi d'un espace, suivi de la chaîne  $S$ . La longueur de la chaîne  $S$  sera toujours d'au moins un et pas plus de 20 caractères. Tous les caractères font partie de l'ensemble des caractères ci-dessus.

### Sortie Standard

Pour chaque jeu de données il y a une ligne de sortie. Il contient le numéro du jeu de données,  $N$ , suivi d'un espace unique qui est ensuite suivi par la nouvelle chaîne  $T$ , qui est constitué de chaque caractère dans  $S$  répété  $R$  fois.

<b>Exemple d'entrée</b>	<b>Exemple de sortie</b>
-------------------------	--------------------------

	Concours National de Programmation Togo 2014 « La bataille des cerveaux »	Campus Numérique Francophone 30 Mai 2014 (C, C++, JAVA)
---	---	---

2 1 3 ABC 2 5 /HTP	1 AAABBBCCC 2 //// /HHHHHTTTTTPPPPP
--------------------------	--

## Problème G : Jeu de Dekon (Ballon violet)

Le jeu de Dekon est un jeu simple généralement joué par deux joueurs. Une version du jeu demande à chaque joueur de choisir une séquence unique de trois pièces, comme PILES ou HEADS, FACES ou TAILS, PILES ou HEADS (HTH). Une pièce de monnaie est lancée séquentiellement un certain nombre de fois jusqu'à ce que un ou deux séquences apparaisse. Le joueur qui a choisi la première séquence apparue gagne le jeu.

Pour ce problème, vous allez écrire un programme qui implémente une variation sur un jeu de Dekon. Vous allez lire une séquence de 40 lancers de pièces et déterminer combien de fois chaque séquence de trois pièces apparaît. Évidemment, il y a huit de ces séquences de trois pièces: TTT, TTH, THT, THH, HTT, HTH, HHT et HHH. Les séquences peuvent se chevaucher. Par exemple, si tous les 40 lancers de pièces sont des têtes, alors la séquence HHH apparaît 38 fois.

### Entrée Standard

La première ligne de l'entrée contient un seul entier **P**, ( $1 \leq P \leq 1000$ ), qui est le nombre de jeux de données à suivre. Chaque jeu de données se compose de 2 lignes. La première ligne contient le numéro de jeu de données **N**. La deuxième ligne contient la séquence de 40 lancers de pièces

Chaque lancer est représenté comme la lettre majuscule H ou la lettre majuscule T, pour pile ou face, respectivement. Il n'y aura pas d'espace sur les lignes d'entrée.

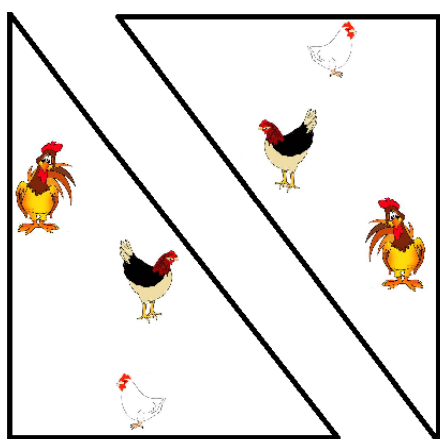
### Sortie Standard

Pour chaque jeu de données, il y a une ligne de sortie. Il contient le numéro de jeu de données suivi d'un espace, suivi du nombre d'occurrences de chaque séquence de trois pièces de

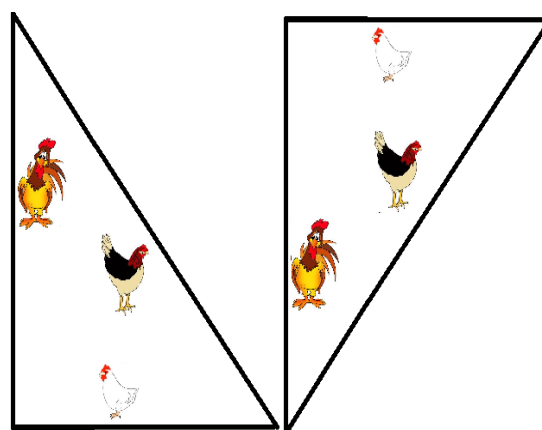
<b>Exemple d'entrée</b>	<b>Exemple de sortie</b>
4	1 0 0 0 0 0 0 0 38
1	2 38 0 0 0 0 0 0 0
HHH	3 4 7 6 4 7 4 5 1
2	4 6 3 4 5 3 6 5 6
TTT	
3	
HHTTTTHHTTTHTHHTHHTTHTTTTHHHTHTTHTTHTTTHTH	
4	
HTHTHHHTHHHTHTHHHHHTTTHTTTTTTHHTTTTHTHHHNT	

## Problème H : Tri Graphs (Ballon rose)

La fermière de poulets Afi est entrain d'acquérir trois nouveaux poulets : Lucy, Charlie et CC. Elle veut construire un poulailler afin que chaque poulet ait sa propre vue, dégagée sur la campagne. Le poulailler aura trois côtés droits; cela donnera à chaque poulet son propre côté de sorte qu'il peut aller et venir sans interférer avec les autres poulets. Afi trouve un rouleau de fil de poulailler (grillage) dans la grange qui est exactement long de  $N$  pieds. Elle veut savoir combien de façons différentes, elle peut faire un poulailler à trois côtés de telle sorte que chaque côté ait un nombre entier de pieds, et qu'elle utilise le rouleau entier de grillage. Différentes rotations du même poulailler sont les mêmes, cependant, Les reflets d'un poulailler peuvent être différents (voir ci-dessous).



*Même*



*Différent*

### Entrée Standard

La première ligne de l'entrée contient un seul entier  $P$ , ( $1 \leq P \leq 1000$ ), qui est le nombre de jeux de données à suivre. Chaque jeu de données doit être traité de façon identique et indépendante.

Chaque jeu de données contient une seule ligne d'entrée. Il contient le numéro de jeu de données  $K$  et la longueur du rouleau de clôture  $N$ , ( $3 \leq N \leq 10000$ ).

## Sortie Standard

Pour chaque jeu de données, il y a une seule ligne de sortie. Il contient le numéro de jeu de données **K** suivi d'un espace, qui est alors suivi par un entier qui est le nombre total de différentes configurations de poulaillers à trois côtés, qui peuvent être faites en utilisant l'ensemble de rouleau de grillage.

Exemple d'entrée	Exemple de sortie
5	1 2
1 3	2 5
2 11	3 4
3 12	4 392
4 100	5 4165834
5 9999	