

# Lëtzebuenger Informatiksolympiad 2015

## Qualifikationsronn

*Il est impératif que vos programmes soient des applications consoles et que leurs sorties respectent le format demandé.*

### Problème I

### Plébiscite

#### Tâche

Un pays imaginaire est subdivisé en circonscriptions électorales. Dans chaque circonscription les électeurs effectuent leurs votes dans des bureaux. Chaque bureau valide uniquement les voix du candidat gagnant, les voix des autres candidats étant donc ignorées. Nous voulons déterminer dans chaque circonscription l'éventuel candidat plébiscité (élu avec une très grande majorité). Nous considérons comme candidat plébiscité celui qui a pu faire valider au total plus que le double des voix validées pour tout autre candidat de la même circonscription.



#### Entrée du programme

##### Principe

La première ligne contient un entier représentant  $C$ , le nombre de circonscriptions.

Pour chaque circonscription on a :

- une ligne avec deux entiers désignant respectivement le nombre de candidats  $CA$  et le nombre de bureaux  $B$ ,
- $B$  lignes représentant chacune un bureau et contenant deux entiers désignant respectivement le numéro du candidat et le nombre de voix obtenues dans le bureau.

##### Restrictions

- le nombre de circonscriptions  $C$  :  $1 \leq C \leq 100$  ;
- le nombre de bureaux  $B$  d'une circonscription :  $1 \leq B \leq 1000$  ;
- le nombre de candidats  $CA$  d'une circonscription :  $2 \leq CA \leq 50$ .

##### Exemple d'exécution

Le programme traitera deux circonscriptions, avec respectivement cinq et quatre bureaux.

```
2
2 5
1 100
1 300
2 80
2 150
1 400
3 4
3 90
2 50
1 100
2 80
```

## Sortie du programme

Pour chaque circonscription le programme affiche :

- dans une première ligne le numéro de la circonscription ;
- dans une deuxième ligne le numéro du candidat plébiscité ou à défaut le texte « Aucun candidat ».

### Exemple d'exécution (correspondant aux données de l'exemple ci-dessus)

```
Circonscription 1:  
1  
Circonscription 2:  
Aucun candidat
```

Dans la première circonscription, le candidat 1 a reçu en tout 800 voix, le candidat 2 seulement 230.

Comme le candidat 1 a reçu plus du double des voix du candidat 2, le candidat 1 est plébiscité.

Dans la deuxième circonscription, le candidat 1 a reçu en tout 100 voix, le candidat 2 a reçu 130 voix et le candidat 3 a reçu 90 voix. Comme aucun des candidats n'a plus que le double des voix des autres candidats, aucun d'eux n'est plébiscité.

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

## Problème II

## Numérologie

### Tâche

La numérologie est un ensemble de croyances et de pratiques fondées sur l'attribution de propriétés à des nombres, propriétés variables selon le contexte (dépendant par exemple de la source alphabétique d'un mot, latin, grec, copte, hébreu, etc.). La numérologie est une pseudo-science. Chaque lettre est représentée par une valeur numérique, indiquant ce que les numérologues appellent sa fréquence vibratoire<sup>1</sup>.



Votre travail consiste à calculer la valeur numérique d'un texte connaissant les fréquences vibratoires de caractères le composant.

### Entrée et sortie du programme

#### Entrée

- La première ligne contient le nombre de traitements à effectuer. Chaque traitement débute par une ligne indiquant le nombre  $k$  de fréquences vibratoires. Les  $k$  lignes suivantes forment une table qui contient des paires (caractère, fréquence vibratoire), ces deux éléments étant séparés par du « whitespace ». L'ensemble de paires est suivi par un entier  $N$  désignant le nombre de lignes de texte à analyser. Chaque ligne de texte peut contenir jusqu'à 20000 caractères. Un caractère ne faisant pas partie de la table a comme fréquence vibratoire 0.

<sup>1</sup> <http://fr.wikipedia.org/wiki/Numérologie>

**Restrictions**

- $k$  est un entier tel que  $0 < k \leq 200$ .
- $N$  est un entier tel que  $0 < N \leq 200000$ .

**Sortie**

- Le programme affiche sur une ligne la valeur numérique du texte représentant la somme des fréquences vibratoires de tous les caractères composant le texte.

**Exemple d'exécution****Entrée**

```
2
3
a 1
b 2
c 3
1
```

La numérologie est un ensemble de croyances et de pratiques fondées sur l'attribution de propriétés

```
5
1 1
, 2
. 3
a 4
L 5
2
```

La numérologie est un ensemble de croyances et de pratiques fondées sur l'attribution de propriétés  
12\*10=120.

**Sortie**

```
14
27
```

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

**Problème III****Activités de week-end****Tâche**

Une commune propose des activités de weekend pour les enfants résidents. La personne chargée de l'organisation a décidé qu'elles sont accessibles exclusivement à des groupes formés d'au moins cinq enfants dont le prénom débute par la même lettre.



Vous devez écrire un programme qui indique la liste des lettres pour lesquelles il y a au moins cinq enfants inscrits.

**Entrée du programme****Principe**

La première ligne contient le nombre d'enfants  $N$  suivi de  $N$  lignes comprenant chacune un prénom  $P$  d'enfant.

**Restrictions**

- le nombre  $N$  d'enfants inscrits au total est compris entre 1 et 150 :  $1 \leq N \leq 150$  ;
- un prénom  $P$  d'enfant est formé d'au plus 30 caractères exclusivement alphabétiques ('a' à 'z' et 'A' à 'Z') et non accentués. Le premier caractère du prénom est toujours écrit en majuscules.

**Exemple d'exécution 1**

12  
Magalie  
Lucas  
Lisa  
Marie  
Melanie  
Lazar  
Mike  
Magalie  
Lana  
Lejla  
Maurice  
Paul

**Exemple d'exécution 2**

13  
Yannick  
Nicolas  
Jenny  
Fabio  
Fabien  
Vincent  
Rachel  
Anastasia  
Lucas  
Axelle  
Melanie  
Pascal  
Steve

**Sortie du programme****Principe**

On affiche le texte « Lettres retenues: » suivi d'un espace et d'une séquence - classée par ordre alphabétique - des caractères pour lesquels il y a au moins cinq enfants dont le prénom commence par le caractère en question. À défaut on indique « / ».

Les exemples ci-dessous résultent des données en entrée indiquées plus haut.

**Exemple d'exécution 1**

Lettres retenues: LM

**Exemple d'exécution 2**

Lettres retenues: /

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

**Problème IV****Entiers et fractions****Tâche**

Beaucoup de calculatrices actuelles peuvent afficher le résultat d'un calcul sous forme de fraction. Votre travail sera plus simple, il se limite à afficher le résultat exact  $r$  du calcul d'une moyenne  $m$  de nombres entiers. La moyenne  $m$  doit être affichée ainsi :



$r$  si  $m$  est un entier

$r \frac{n}{d}$  si  $m$  est un nombre fractionnaire et  $|m| > 1$

$\frac{n}{d}$  si  $m$  est un nombre fractionnaire et  $|m| < 1$

Si  $m$  est un entier négatif, le signe moins '-' suivi par exactement un espace précéderont le résultat.

Dans la fraction  $\frac{n}{d}$ ,  $n$  et  $d$  sont premiers entre eux, c.-à-d. leur plus grand commun diviseur est égal à 1.

La barre de fraction doit être composée d'autant de caractères '-' que de chiffres composant le dénominateur  $d$ . Le numérateur  $n$  doit toujours être aligné à droite.

## Entrée et sortie du programme

### Entrée

- Les données nécessaires au calcul d'une moyenne tiennent sur une ligne. Une ligne débute par un entier  $N$  désignant combien de nombres  $nb_j$  sont à additionner pour ce calcul de moyenne. Les  $N$  entiers suivants  $nb_j$  sont séparés par du « whitespace ». La ligne qui débute par 0 termine le programme.

### Restrictions

- $N$  est un entier tel que  $1 \leq N \leq 200$ .
- $nb_j$  avec  $1 \leq j \leq N$  sont des entiers compris entre -1000 et 1000.

### Sortie

- Le programme affiche pour chaque cas le numéro du cas suivi de la moyenne en respectant les règles d'affichage données ci-dessus.

## Exemples d'exécution

### Entrée

```
5 1 2 3 4 5
5 2 4 6 10 12
10 1 -1 1 -1 1 -1 -1 1 1 1
3 -1 -2 -3
3 -1 -2 -4
11 1 2 3 4 5 6 7 8 9 10 97
0
```

### Sortie

```
Moyenne 1:
 3
Moyenne 2:
 4
6-
 5
Moyenne 3:
 1
-
 5
Moyenne 4:
- 2
Moyenne 5:
 1
- 2-
 3
Moyenne 6:
 9
13--
 11
```

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