be-OI 2011 **Demi-finale** **Solutions** 16 Février 2011	**Solutionnaire** **Demi-finales 2011** **Catégories secondaire et supérieur**	

Olympiades belges d'Informatique

Question 1 – Échauffement

Q1(a) n

Une façon de résoudre ce problème était de l'exécuter sur papier avec $n = 3$. Une autre façon était de se rendre compte que ce calcul revient à faire la somme de tous les éléments d'une matrice carrée $n \times n$, dont les éléments supérieurs à la diagonale principale valent -1 et tous ceux sur ou sous la diagonale valent 1. On voit rapidement dans une telle matrice que tous les éléments hors de la diagonale s'annulent. Il reste donc la diagonale, qui contient bien entendu n éléments.

Q1(b) 31

Il fallait bien entendu faire attention à continuer l'itération lorsque i valait 15 et à ne pas se tromper dans les calculs.

Q1(c) `not (a % 2 = 0 or b % 2 = 0)`

En utilisant le symbole modulo (%) expliqué en première page, on pouvait écrire "a est impair" comme "`a % 2 = 1`" ou encore "`not (a % 2 = 0)`". L'expression donnée pouvait donc s'écrire "`not (a % 2 = 0) and not (b % 2 = 0)`", ce qui est équivalent à "`not (a % 2 = 0 or b % 2 = 0)`".

Question 2 – Mélange de cartes

Q2(a) secondaire `3 4 1` **Q2(a) supérieur** `3 5 2`

Q2(b) secondaire `2 3 4` **Q2(b) supérieur** `2 4 5`

Q2(c) secondaire `4 3 2` **Q2(c) supérieur** `4 3 1`

Pour résoudre ce problème, il suffit de toujours essayer de placer sur C le plus grand chiffre parmi toutes les cartes de A et la première (la plus à gauche) de B. Si le plus grand chiffre se trouve sur la table A, mais pas en première position, il faut effectuer des opérations Y jusqu'à ce que ce soit le cas. Lorsque ce chiffre est en première position de A ou B, il suffit de le déplacer en C. Cet algorithme doit être appliqué tant que toutes les cartes ne sont pas sur C.

Question 3 – Vive le vent ...

Cette question contenait une grosse erreur non volontaire. L'algorithme proposé devait renvoyer b et non a afin de pouvoir résoudre tous les scénarios. Afin de ne pas pénaliser ceux n'ayant pas trouvé de solution à cause de cette erreur, cette question n'a pas été comptabilisée pour les candidats pour lesquels cela aurait fait diminuer la moyenne. Nous avons également été plus souples dans la correction pour les mêmes raisons. Les solutions ci-dessous sont correctes pour le cas où l'algorithme retourne b.

Q3(a) `a != b`

Q3(b) `( a + b ) / 2`

La solution attendue était une recherche dichotomique qui consiste à toujours prendre la prochaine mesure à mi-distance parmi les valeurs encore possibles. Ce qui permet à chaque itération de réduire le nombre de valeurs possibles de moitié.

Question 4 – Ceci n'est pas un sudoku

Q4(a) secondaire 7 (ou 6) **Q4(a) supérieur** 4

Q4(b) secondaire 7 **Q4(b) supérieur** 10

Q4(c) secondaire 5 **Q4(c) supérieur** 12

Q4(d) secondaire 5 **Q4(d) supérieur** 10

Dû à une imprécision, il était possible de comprendre l'énoncé de deux manières différentes. Selon l'interprétation, il était donc possible de prendre en compte la case en haut à gauche dans le calcul ou non. Nous avons accepté les deux interprétations, c'est ce pourquoi il y avait deux réponses possibles à la Q4(a) pour le secondaire.

Il y avait plusieurs stratégies pour résoudre ce problème. L'une d'entre elle consistait à commencer par calculer le score pour arriver à chacune des cases de la première colonne et ligne, et d'ensuite calculer chacune des cases dont le score de la case à gauche et au dessus est connu en utilisant le plus petit des deux scores pour continuer. Cette stratégie est ensuite optimisable afin de ne pas avoir à calculer le score pour l'entièreté des cases.

Question 5 – Où organiser la Finale ?

Q5(a) Namur

Q5(b) secondaire Mons **Q5(b) supérieur** Anvers

Q5(c) `s + distance[i][j]`

Q5(d) `s / ( N - 1 )`

Q5(e) `rang[k][1] > m`

Le principe du programme consiste à, pour chaque centre :
- calculer la somme (variable s) de ses distances aux autres centres,
- calculer la moyenne en divisant cette somme par le nombre d'autres centres
- de trouver la place dans la liste ordonnée où insérer cet élément en incrémentant k tant que l'élément placé actuellement en k a une moyenne plus élevée.

Question 6 – Le jeu des craies

Q6(a) `( remaining - 1 ) % 4`

ou encore **Q6(b)** `remaining = 30` (ou `last = 0`) **Q6(c)** `1` **Q6(d)** `4 - last`

Ce jeu est l'une des variantes du jeu de Nim. Pour gagner, il faut laisser à l'adversaire 1 craie, ou 5 craies (parce qu'il vous en laissera 2, 3 ou 4, ce qui vous permettra de lui en laisser 1), ou 9 craies (parce qu'il vous en laissera 6, 7 ou 8, ce qui vous permettra de lui en laisser 5), ou 13, 17, 21, 25 ou 29. Il faut donc, pour être certain de gagner, laisser à l'adversaire $1 + 4x$ (x entier positif) craies. Vu que l'on vous dit que la situation actuelle permet toujours à Martin de gagner, "`( remaining - 1 ) % 4`" est la solution et ne vaudra jamais zéro.

Notez que la solution (a) n'a été trouvée que par 17 des 252 participants et que la (a-b-c) a été trouvée par 20 autres participants.

Question 7 – Parlez-vous la *Zorglangue* ?

Q7(a)

```
for (i ← 0 to (fin − debut + 1)/2 − 1 step 1)
{
    tmp ← phrase[debut + i]
    phrase[debut + i] ← phrase[fin − i]
    phrase[fin − i] ← tmp
}
```

ou encore

```
for (i ← debut to (fin + debut − 1)/2 step 1)
{
    tmp ← phrase[i]
    phrase[i] ← phrase[fin + debut − i]
    phrase[fin + debut − i] ← tmp
}
```

Notez que si le mot à inverser avait une taille impaire, il était inutile d'échanger le caractère central avec lui-même.

Q7(b) `phrase[i] = ' ' or phrase[i] = '.'` (ou encore `phrase[i] = ' ' or i = n - 1`)

Q7(c) `i - 1`

Q7(d) `wordstart ← i + 1`

Question 8 – Anagrammes numériques

Solution utilisant un tri, sans bonus :

```
tri(a,n)
tri(b,n)

for (i ← 0 to n − 1 step 1)
{
    if (a[i] ≠ b[i])
    {
        return false
    }
}
return true
```

Solution pour obtenir le bonus :

```
count ← newarray (10)

for (i ← 0 to n − 1 step 1)
{
    count[a[i]] ← count[a[i]] + 1
    count[b[i]] ← count[b[i]] − 1
}

for (i ← 0 to 9 step 1)
{
    if (count[i] ≠ 0)
    {
        return false
    }
}
return true
```

De nombreuses autres solutions étaient également possibles.

Question 9 – Préparer mon planning (uniquement supérieur)

Q9(a) $f - d$

Q9(b) $premier = -1 \ \ or \ \ f \leq debut$

Q9(c) $i \neq -1$

Q9(d) $suivant[place]$

Q9(e) $deb\_i - f$