



**Le Centre d'éducation
en mathématiques
et en informatique**

*Concours
canadien
d'informatique
2014*

*Niveau
intermédiaire*

Commanditaire :

**WATERLOO
MATHEMATICS**

Concours canadien d'informatique — niveau intermédiaire

Directives à l'intention des participantes et des participants

1. Vous pouvez participer à un concours seulement. Pour participer au concours de niveau supérieur, il faut choisir l'autre trousse de problèmes.
2. Sur le formulaire **Information à l'intention des élèves**, indiquez que vous participez au concours de niveau **intermédiaire**.
3. Vous avez trois (3) heures pour accomplir le travail.
 - si votre enseignant note votre copie, toutes les entrées se font par le biais du clavier ;
 - si vous utilisez le correcteur CCC en ligne (On-line CCC grader), toutes les entrées se font par entrées standard ;
 - toutes les sorties se font par sorties standard, (c.-à-d. par l'écran).Aucune sollicitation n'est requise. Les sorties doivent être **IDENTIQUES** à celles des exemples de sorties, par rapport à l'ordre, aux espaces, etc.
4. Vous devez faire votre propre travail. Les tricheurs seront punis sévèrement.
5. Il est interdit de faire appel à des caractéristiques auxquelles le juge (votre enseignant ou le correcteur CCC en ligne) n'a pas accès pendant l'évaluation de votre programme. Si vous utilisez le correcteur CCC en ligne, inscrivez bien le type et la version du compilateur utilisé par votre langage de programmation.
6. Vous pouvez consulter des livres et du matériel écrit. Tout matériel susceptible d'être lu électroniquement (par exemple un programme que vous avez écrit) est *interdit*. Cependant, vous pouvez faire appel aux bibliothèques reconnues pour vos langages de programmation ; par exemple, STL pour C++, java.util.*, java.io.* et autres pour Java, et ainsi de suite.
7. Vous devez vous limiter aux applications de programmation ordinaires (éditeurs, compilateurs, débogueurs). Toutes les autres applications sont **interdites**. Leur utilisation entraînera une disqualification.
8. Utilisez des noms de fichiers qui sont propres à chaque problème : `j1.pas` ou `j1.c` ou `j1.java` (ou tout autre suffixe de fichier approprié) pour le problème J1. Si vous utilisez le correcteur CCC en ligne, suivez les directives qui y sont données quant aux noms de fichiers, tout en portant une attention particulière aux noms de fichiers pour les programmes Java.
9. Votre programme sera exécuté avec des scénarios d'essais différents de ceux qui figurent comme exemples. Assurez-vous de vérifier votre programme au moyen d'autres scénarios d'essais. Assurez-vous d'avoir un code aussi performant que possible par rapport au temps. Vous disposerez d'au plus 5 secondes d'exécution par scénario d'essai.
10. Consultez le site web du CCI à la fin du mois de mars pour connaître votre classement dans ce concours et pour connaître le nom des gagnants. Voici l'adresse :

www.cemc.uwaterloo.ca/ccc

Problème J1 : Types de triangles

Description du problème

Vous avez de la difficulté à vous rappeler les divers types de triangles. Vous écrivez un programme pour vous aider.

Votre programme lira trois nombres, soit la mesure des trois angles (exprimés en degrés).

- Si les trois mesures égalent 60, la sortie sera `Equilateral`.
- Si les trois mesures ont une somme de 180 et si exactement deux des mesures sont égales, la sortie sera `Isocele` si le programme est noté par l'enseignant ou `Isosceles` si le programme est noté par le correcteur en ligne.
- Si les trois mesures ont une somme de 180 et si elles sont différentes les unes des autres, la sortie sera `Scalene`.
- Si les trois mesures n'ont pas une somme de 180, la sortie sera `Erreur` si le programme est noté par l'enseignant ou `Error` si le programme est noté par le correcteur en ligne.

Précisions par rapport aux entrées

L'entrée comportera trois entiers, soit un par ligne.

Chaque entier sera supérieur à 0 et inférieur à 180.

Précisions par rapport aux sorties

Si le programme est noté par l'enseignant, la sortie sera exactement un des mots suivants sur une ligne : `Equilateral`, `Isocele`, `Scalene` ou `Erreur`

Si le programme est noté par le correcteur en ligne, la sortie sera exactement un des mots suivants sur une ligne : `Equilateral`, `Isosceles`, `Scalene` ou `Error`

Exemple d'entrée 1

```
60
70
50
```

Sortie pour l'exemple d'entrée 1

```
Scalene
```

Exemple d'entrée 2

```
60
75
55
```

Sortie pour l'exemple d'entrée 2

```
Erreur (si noté par l'enseignant)
```

```
ou
```

```
Error (si noté par le correcteur en ligne)
```

Problème J2 : Résultats du vote

Description du problème

La chanteuse A et la chanteuse B participent à la finale d'un concours de chant. La gagnante est choisie selon le résultat d'un vote.

Vous devez compter les votes et déterminer le résultat.

Précisions par rapport aux entrées

L'entrée comporte deux lignes. La première ligne contient V ($1 \leq V \leq 15$), soit le nombre total de votes reçus. La deuxième ligne contient une suite de V caractères, chaque caractère étant un A ou un B , indiquant ainsi un vote pour A ou pour B .

Précisions par rapport aux sorties

La sortie sera une des trois possibilités suivantes :

- Si A reçoit plus de votes que B , la sortie sera `A` ;
- Si B reçoit plus de votes que A , la sortie sera `B` ;
- Si A et B reçoivent le même nombre de votes, la sortie sera `Egalite` si le programme est noté par l'enseignant ou `Tie` s'il est noté par le correcteur en ligne.

Exemple d'entrée 1

```
6
ABBABB
```

Sortie pour l'exemple d'entrée 1

```
B
```

Exemple d'entrée 2

```
6
ABBABA
```

Sortie pour l'exemple d'entrée 2

```
Egalite (si noté par l'enseignant)
ou
Tie (si noté par le correcteur en ligne)
```

Problème J3 : Deux dés

Description du problème

Antonia et David participent à un jeu.

Au départ, chaque joueur a 100 points.

Le jeu comporte un certain nombre de rondes et chaque joueur utilise un dé régulier à six faces. Dans une ronde, chaque joueur jette son dé une fois et on note le nombre sur la face supérieure de chaque dé. Le joueur qui obtient le nombre le plus bas perd un nombre de points égal au plus élevé des deux nombres. Si les deux dés indiquent le même nombre, aucun joueur ne perd des points.

Vous devez écrire un programme qui indique le nombre de points de chaque joueur à la fin d'une joute.

Précisions par rapport aux entrées

La première ligne d'entrée contient un entier n ($1 \leq n \leq 15$), soit le nombre de rondes dans la joute. Chacune des n lignes suivantes contient deux entiers : le nombre obtenu sur le dé d'Antonia dans cette ronde, suivi d'une espace, suivi du nombre obtenu sur le dé de David dans cette ronde. Chaque nombre sera un entier de 1 à 6.

Précisions par rapport aux sorties

La sortie comporte deux lignes. La première ligne contient le nombre de points qu'il reste à Antonia après que toutes les rondes ont été jouées. La deuxième ligne contient le nombre de points qu'il reste à David après que toutes les rondes ont été jouées.

Exemple d'entrée

```
4
5 6
6 6
4 3
5 2
```

Sortie pour l'exemple d'entrée

```
94
91
```

Explication de la sortie pour l'exemple d'entrée

David gagne la première ronde. Donc, Antonia perd 6 points. Dans la deuxième ronde, il y a égalité. Aucun point n'est perdu. Antonia gagne la troisième ronde. Donc, David perd 4 points. Antonia gagne la quatrième ronde. Donc, David perd 5 points. En tout, Antonia a perdu 6 points et David a perdu 9 points.

Problème J4 : Invitation à une fête

Description du problème

Vous préparez une fête, mais vous n'avez pas assez de place pour inviter tous vos amis. Pour déterminer les amis que vous inviterez, vous utilisez la méthode suivante qui est exempte d'émotions.

Attribuez un numéro $1, 2, \dots, K$ à chacun de vos K amis et placez les numéros dans une liste en ordre croissant. Il y aura ensuite m rondes. Dans chaque ronde, vous utilisez un nombre pour déterminer quels numéros d'amis seront enlevés de la liste.

Dans les rondes, les nombres r_1, r_2, \dots, r_m seront utilisés. Dans la ronde i , vous enlevez les numéros d'amis dont la position actuelle en liste est un multiple de r_i (c'est-à-dire les positions $r_i, 2r_i, 3r_i, \dots$). Dans chaque ronde, la nouvelle liste commence par la position 1.

Votre programme donnera en sortie les numéros d'amis qui sont encore en liste à la fin de toutes les rondes.

Précisions par rapport aux entrées

La première ligne d'entrée contient un entier K ($1 \leq K \leq 100$). La deuxième ligne contient un entier m ($1 \leq m \leq 10$), soit le nombre de rondes. Chacune des m lignes suivantes contient un entier. La $i^{\text{ème}}$ de ces lignes ($1 \leq i \leq m$) contient r_i ($2 \leq r_i \leq 100$), ce qui indique que les numéros d'amis dont la position est un multiple de r_i doivent être enlevés de la liste.

Précisions par rapport aux sorties

La sortie contient les numéros d'amis qui n'auront pas été enlevés. Ils doivent paraître en ordre croissant, un numéro par ligne.

Exemple d'entrée

```
10
2
2
3
```

Sortie pour l'exemple d'entrée

```
1
3
7
9
```

Explication de la sortie pour l'exemple d'entrée

Au départ, les numéros d'amis sont 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. Il y a deux rondes. Dans la première ronde, on enlève les numéros dans les positions paires (les positions 2, 4, 6, 8 et 10) et à la fin de cette ronde, la liste comporte les numéros d'amis 1, 3, 5, 7, 9. Dans la deuxième ronde, on enlève les numéros dans les positions qui sont des multiples de 3 (la position 3 qui contient le numéro 5). À la fin de cette ronde, la liste comporte les numéros d'amis 1, 3, 7, 9.

Problème J5 : Attribution des partenaires

Description du problème

Le CEMI organise un atelier. Dans une activité, les participants doivent travailler en paires. Le comité organisateur décide de choisir les partenaires maintenant, avant le début de l'atelier. Vous devez déterminer si le choix de partenaires est *cohérent*, c'est-à-dire que si A est le partenaire de B, alors B doit être le partenaire de A et aucun participant n'est son propre partenaire.

Précisions par rapport aux entrées

L'entrée comporte trois lignes. La première contient un entier N ($1 < N \leq 30$), soit le nombre d'élèves inscrits. La deuxième ligne contient le prénom des N élèves, séparés d'une espace. (Les noms ne contiennent que des lettres majuscules ou minuscules et tous les prénoms sont distincts.) La troisième ligne contient les mêmes N prénoms, dans un certain ordre, séparés d'une espace.

Les positions des noms dans les deux dernières lignes indiquent les partenaires : le $i^{\text{ième}}$ nom dans la deuxième ligne est le partenaire du $i^{\text{ième}}$ nom dans la troisième ligne.

Précisions par rapport aux sorties

Si la notation est faite par l'enseignant, la sortie est `bon` si les deux listes donnent un choix de partenaires cohérent et `mal` si elles donnent un choix qui n'est pas cohérent.

Si la notation est faite par le correcteur en ligne, la sortie est `good` si les deux listes donnent un choix de partenaires cohérent et `bad` si elles donnent un choix qui n'est pas cohérent.

Exemple d'entrée 1

4

Ada Alan Grace John
John Grace Alan Ada

Sortie pour l'exemple d'entrée 1

`bon` (si noté par l'enseignant) ou `good` (si noté par le correcteur en ligne)

Explication de la sortie pour l'exemple d'entrée 1

Ada et John sont partenaires ; Alan et Grace sont partenaires. Le choix est cohérent.

Exemple d'entrée 2

7

Rich Graeme Michelle Sandy Vlado Ron Jacob
Ron Vlado Sandy Michelle Rich Graeme Jacob

Sortie pour l'exemple d'entrée 2

`mal` (si noté par l'enseignant) ou `bad` (si noté par le correcteur en ligne)

Explication de la sortie pour l'exemple d'entrée 2

Graeme a pour partenaire Vlado, mais Vlado a pour partenaire Rich. Le choix n'est pas cohérent. De plus, Jacob est son propre partenaire, ce qui est incohérent.