

DA IEEE CODE

Compétition de programmation – hiver 2017

Énoncé de la compétition

23 mars 2017

Branche étudiante de l'IEEE de l'Université Laval

Contents

1	Introduction	3
2	Rèlements	4
2.1	Généralités	4
2.2	Composition des équipes	4
2.3	Inscription	4
2.4	Équipement, ressources permises et nécessaires	5
2.5	Déroulement de la compétition	5
2.6	Remise et évaluation	5
2.7	Triche, plagiat et disqualification	6
2.8	Remise des prix	7
2.9	Vous avez des questions ?	7
3	Organisation	8
4	Épreuves	9
4.1	Problème 1 : Réorganiser les noms des pokémons (Langage: au choix, Pondération : 10%)	9
4.2	Problème 2 : Zee Pokédex (Langage: au choix, Pondération : 20%)	10
4.3	Problème 3 : Décoder le génome des pokémons (Langage : C++, Pondération : 10%)	11
4.4	Problème 4 : Sing me the song of your people (Langage : c++, Pondération : 20%)	12
4.5	Problème 5 : I challenge you (Language : c++, pondération : 40%)	13

1 Introduction

La sixième édition du DA IEEE CODE se déroulera le jeudi 23 mars entre 13h30 et 17h30 au local 0103 du pavillon Adrien-Pouliot. Cette compétition est organisée par la branche étudiante de l'IEEE.

Cette compétition de programmation prendra la forme d'une compétition composée de plusieurs défis de programmation. Les détails des défis et des tâches à accomplir seront expliqués en détail au début de la compétition à 13h30.

La compétition s'effectue en équipe de deux. Afin d'être déclaré vainqueur, une équipe devra résoudre les défis et marquer le maximum de points selon les barèmes expliqués plus loin.

Le présent document présente les règles de fonctionnement de la compétition ainsi que son déroulement.

2 Règlements

2.1 Généralités

- Tous les règlements se doivent d'être appliqués dans le but unique que chaque participant ait un maximum de plaisir ainsi qu'une chance égale de gagner. Les juges appliqueront les règlements dans la plus grande convivialité possible.
- Tous les documents de la compétition seront en français. Par convention, le code fourni a été écrit en anglais. Il est toutefois permis à un compétiteur d'utiliser le langage de son choix dans son livrable.
- Le masculin est utilisé de façon générique pour simplifier le texte ; femmes et hommes sont également acceptés pour la compétition!

2.2 Composition des équipes

- Chaque équipe est constituée d'un étudiant en 1ère ou 2e année de son programme d'études ainsi que d'un second étudiant de son programme d'étude, préférablement de 3e ou 4e année (Une exception est appliquée aux étudiants en génie électrique qui comptent comme des étudiants de 1ère et 2e année). Pour les personnes au cheminement particulier, un 1ère année est quelqu'un qui possède moins de 30 crédits; un 2e année quelqu'un qui a moins de 60 crédits ; un 3e année quelqu'un qui a moins de 90 crédits et un 4e année quelqu'un qui a moins de 120 crédits.
- Les étudiants doivent être inscrits dans un programme de premier cycle de génie électrique, génie informatique, génie logiciel, informatique ou d'un programme de premier cycle avec suffisamment de similarités avec un de ces derniers.

2.3 Inscription

- Les équipes de deux doivent s'être inscrites pour la compétition via le document suivant : <https://goo.gl/forms/RLr01DbvSSwMiaXV2>
- Les participants acceptent d'être pris en photo durant la compétition. Ces photos pourront être employées à des fins de promotions et publicité pour l'évènement, pour la promotion de la branche étudiante de l'IEEE de l'Université Laval.
- Les gagnants seront photographiés lors de la remise des prix. Ces photographies et leur nom, prénom et programme d'étude seront employés à des fins de promotion et publicités pour l'évènement, pour la promotion de la branche étudiante de l'IEEE de l'Université Laval.
- Le nombre d'équipes est limité par les dimensions de la salle. veuillez-vous inscrire dès que possible. Premier arrivé (et qualifié), premier servi.

- Des inscriptions individuelles sont acceptées. Ces participants seront placés en équipe sur place si nécessaire. Nous ne garantissons pas qu’il soit possible de former des équipes avec les inscrits individuellement et ceux-ci doivent accepter le fait qu’ils risquent de ne pas pouvoir prendre part à la compétition.

2.4 Équipement, ressources permises et nécessaires

- On permet aux compétiteurs d’apporter leur propre ordinateur portable. Le programme devrait tout de même pouvoir s’exécuter sur une station de bureau présente au laboratoire.
- Une machine virtuelle Linux sera fourni aux participants. Libre à eux de l’utiliser ou non mais l’environnement d’évaluation sera le même que celle-ci.
- Des stations de bureau sont aussi fournies en nombre limité.
- L’internet, l’électricité, la lumière, les bureaux et les chaises ainsi que l’oxygène sont fournis sans frais aux participants.
- Les programmes devront être implémentés en Python 3.4 ou dans une version plus récente ainsi qu’en C++.
- Les participants sont libres d’utiliser l’environnement de développement de leur choix. Toutefois, ils doivent être capables d’exécuter leur programme en ligne de commande afin de s’assurer que la correction se fasse de manière appropriée.

2.5 Déroulement de la compétition

- La durée de la compétition est limitée à quatre heures, débutant à 13h30 le 23 mars et se terminant à 17h30 sans faute.
- Au tout début de la compétition, les équipes reçoivent le présent document contenant les règles. Ils reçoivent également le lien vers le dossier comprenant les défis. Un fichier par défi permet d’avoir les détails du programme à effectuer.
- Les présences seront prises à plusieurs reprises durant l’évènement. Des ordinateurs de bureau seront mis à la disposition des compétiteurs, ainsi qu’Internet, des prises électriques, chaises, bureaux, etc.
- Des breuvages et collations seront fournis.

2.6 Remise et évaluation

- Les défis seront évalués par un comité composé d’un ou plusieurs juges, et ce, suivant la compétition. La validité de la réponse pour chaque problème sera évaluée. La qualité du code ainsi que son efficacité seront des critères pris en compte lors de l’évaluation.

- L'évaluation du code sera fait après la compétition.
- La remise doit se faire sur github. Chaque équipe devra se créer une branche et pousser son code sur celle-ci.
- Il est de la responsabilité de chaque équipe d'indiquer comment l'évaluateur doit exécuter son code. Du code qui ne compile pas/ne s'exécute pas se verra attribuer la note zéro.
- Dans le cas d'une égalité, l'équipe qui aura indiquer au juge avoir terminer la compétition en premier se verra attribuer la victoire sur l'autre équipe.

2.7 Triche, plagiat et disqualification

- Il est strictement interdit de copier du code trouvé sur Internet.
- L'application stricte du règlement 2.7.1 résulterait en une compétition de programmation à toutes fins pratiques impossible. Voici donc des lignes directrices en ce qui concerne son application :
 1. Copie d'appels de fonctions et d'exemples provenant de sites de référence sur les langages de programmation : permis (et encouragé).
 2. Copie de petits bouts de code (d'un nombre de lignes se comptant sur les doigts de la main) trouvés sur StackOverflow ou un site similaire : toléré, sujet à interprétation.
 3. Copie de paragraphes entiers de code trouvés sur GitHub, Google Code, SourceForge ou un site similaire : interdit.
 4. Copie de paragraphes entiers de code, en changeant les noms de variables et en déplaçant un peu le code afin de camoufler sa provenance : interdit.
 5. « S'inspirer » de code déjà existant pour produire son programme, au lieu de développer ses propres algorithmes : interdit.

Le jury a le dernier mot quant à l'application de cette règle, et ses décisions sont finales et sans appel. Des règles similaires à celles du plagiat dans les travaux universitaires, ainsi que le gros bon sens seront appliqués.

En contrepartie, l'honnêteté de tous les participants est demandée. Il s'agit d'une compétition de programmation, pas une compétition de copier-coller. Ce n'est pas un cours, vous ne risquez pas l'échec. Des points seront quand même donnés si votre code ne s'exécute pas parfaitement: voir les barèmes propres à chaque défi.

- Un participant violant un ou plusieurs des présents règlements verra automatiquement son équipe disqualifiée. Les décisions du jury à cet égard sont finales et sans appel.

2.8 Remise des prix

Les prix et bénéfices de la compétition sont les suivants :

1. L'équipe en première position recevra un prix de 250\$
2. L'équipe en seconde position recevra un prix de 150\$
3. L'équipe en troisième position recevra un prix de 100\$

Quelques règles concernant les prix :

1. La liste des gagnants sera disponible sur le site de l'IEEE <http://ieee.gel.ulaval.ca>.
2. Les prix aux équipes gagnantes seront remis au maximum deux semaines après la compétition.
3. S'il s'avère impossible à un gagnant d'être présent lors de la remise, il aura 14 jours pour réclamer son prix à compter de la date de la remise des prix. Après quoi, son prix est abandonné et ne sera pas remis à quiconque.

2.9 Vous avez des questions ?

Pour toutes questions ou clarifications, écrivez à julien.morissette.1@ulaval.ca.

Les présents règlements seront interprétés par le jury organisateur, leurs décisions et interprétations seront finales et sans appel.

3 Organisation

La compétition aura lieu pendant une durée de 4h en équipe de 2. Une configuration “minimale” sera exigée; c’est-à-dire que les participants doivent avoir configuré leurs ordinateurs d’ici la compétition. Une machine virtuelle ayant tout le nécessaire d’installé pour la compétition sera aussi fournie aux participants. Si vous n’avez pas d’ordinateur veuillez communiquer avec les organisateurs ils tenteront de vous accommoder avec les ordinateurs présents au 0103.

Pendant la compétition, chaque équipe participante pourra poser des questions au comité organisateur. Celui-ci jugera s’il peut y répondre ou non.

4 Épreuves



Bonjour à vous, dresseurs de Pokemons. C'est le professeur Oak qui vous parle. Vous allez bientôt débiter votre grande aventure et partir parcourir le monde afin de devenir le plus grand dresseur. Normalement, je vous ferais choisir votre propre Pokémon et vous enverrai sur le chemin d'aventure mais j'ai malheureusement quelques problèmes avec votre Pokédex. J'ai cru comprendre que vous aviez appris les voix mystiques de la programmation avant de tous laisser tomber pour devenir un dresseur de pokémon. Peut-être serez-vous en mesure de m'aider. J'ai 5 problèmes pour vous, lorsque vous les aurez réglés, vous pourrez partir à la découverte de votre destinée.

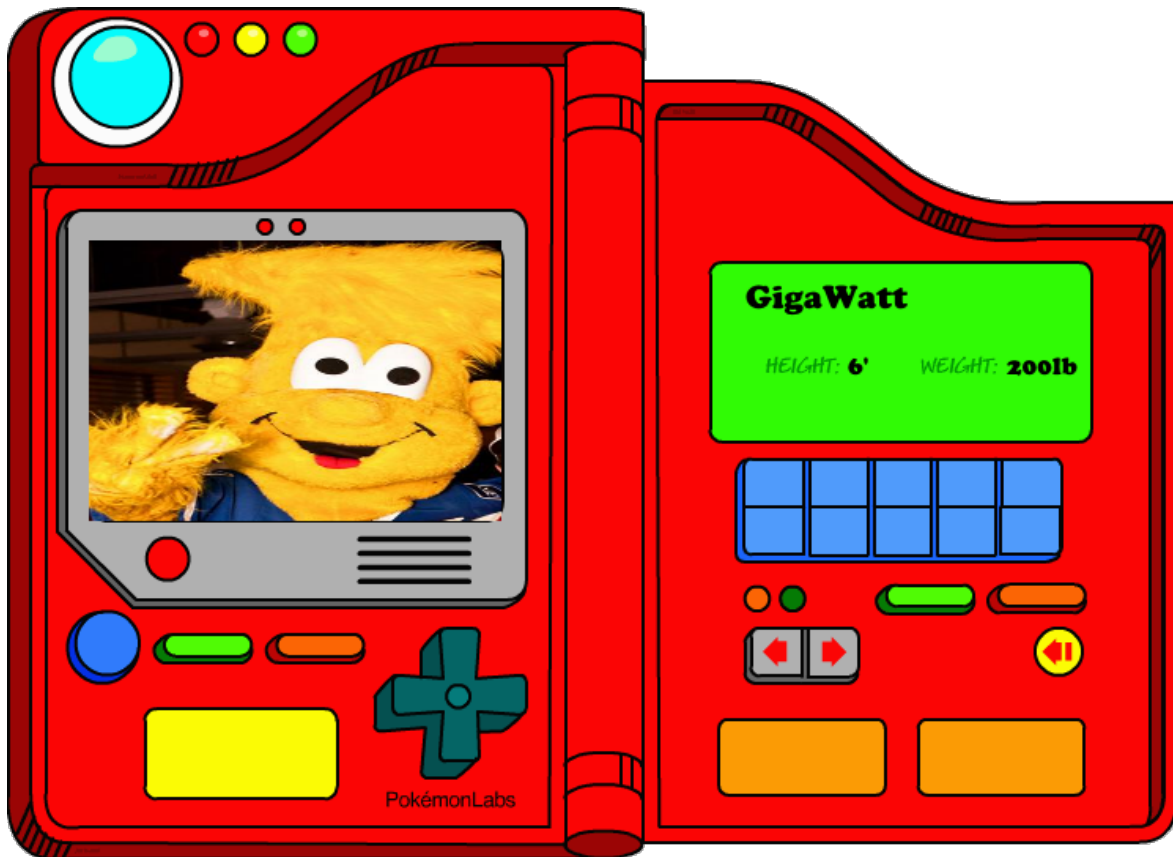
Le repo pour la compétition se trouve ici : <https://github.com/Crom22/daieeetcode2017>.
git

4.1 Problème 1 : Réorganiser les noms des pokémons (Langage: au choix, Pondération : 10%)

Vous voyez c'est un peu gênant, votre pokédex affiche les pokémons mais leurs noms ne font aucun sens et les lettres sont toutes mélangées entre elles. Heureusement, j'ai encore une bonne vieille liste contenant tous les pokémons en format papier. Peut-être qu'avec l'aide de celle-ci vous pourrez remettre de l'ordre dans votre pokédex?

Votre défi est de programmer un "name reorderer". Vous trouverez dans le fichier "Challenge 1 and 2" un fichier contenant la liste des pokémons de la première génération en format csv. Vous trouverez aussi un exemple de noms mélangés dans le fichier `pokemon_names_to_reorder.txt`. Votre programme doit utiliser le fichier csv pour déterminer le nom du pokémon. L'entrée de votre programme est un nom de pokémon mélangé et la sortie de celui-ci est le nom du pokémon ou bien "no pokemon found" si on entre une entrée invalide.

4.2 Problème 2 : Zee Pokédex (Langage: au choix, Pondération : 20%)



Maintenant que vous affichez les noms correctement peut-être que votre pokédex devrait faire plus ?

Votre deuxième défi sera d'ajouter une option de recherche à votre pokédex. Celui-ci devra pouvoir retourner les informations du pokémon correspondant sous la forme suivante : "Nom du Pokemon: His number is Numero du Pokemon and he is of type Son type". Il devra retourner "no pokemon found" si le pokémon est invalide.

4.3 Problème 3 : Décoder le génome des pokémons (Langage : C++, Pondération : 10%)



J'aurais maintenant une petite faveur à vous demander. J'ai ce fossile de Pokémon qui traîne depuis plusieurs années. J'aimerais vraiment pouvoir comparer le génome de Pokémon préhistoriques avec ceux de l'ère moderne. Peut-être pourriez-vous m'aider?

Votre troisième défi sera de coder une fonction C++ qui compte le nombre de "1" dans la représentation binaire d'un long. Vous pourrez tester votre algorithme à l'aide du main.cpp. Sachez que ces exemples sont seulement à titre indicateur, assurez-vous que ça fonctionne dans les cas généraux.

Ex : 5 en binaire = 101 soit 2 "1"

4.4 Problème 4 : Sing me the song of your people (Langage : c++, Pondération : 20%)

The image shows the text "Gotta catch émail!" in a large, blue, stylized font with a thick outline. The text is slanted upwards from left to right. A small "TM" trademark symbol is located at the end of the text.

Ho, j'y pense, j'ai aussi ce texte ancien que je n'arrive pas à comprendre. Peut-être pourriez-vous m'aider en programmant une application me permettant de voir l'occurrence des différents symboles dans le texte?

Votre quatrième défi sera de coder un programme C++ qui compte le nombre d'occurrences de mots dans un fichier txt et sauvegarde le résultat dans un fichier json. Le format du fichier doit correspondre à :

```
{  
  "mot1": 5,  
  "mot2": 3  
}
```

Indication supplémentaire : Les "" sont considérés comme une lettre faisant partie d'un mot. Vous pouvez considérer les parenthèses et autres ponctuations comme un mot à part entières.

4.5 Problème 5 : I challenge you (Language : c++, pondération : 40%)



Ho, je me suis rappelé pendant que vous m'aidiez avec mes antiquités qu'il vous faut une façon de gérer votre équipe de pokémon. Malheureusement, votre pokédex ne semble pas faire l'affaire. Vous pouvez utiliser mon vieux programme de combat. Cependant, même si le menu est encore là il ne semble plus fonctionner. Peut-être serez-vous en mesure de le réparer?

Votre dernier défi sera de programmer, en c++, une application permettant de gérer votre équipe de pokémons. L'application implémente la gestion de pokémons sous la forme d'une liste chaînée circulaire. L'application devra permettre d'ajouter un pokémon dans votre équipe à une position x. D'enlever un pokémon par sa position. D'afficher le nombre de pokémon dans votre équipe. D'afficher le pokémon avec le plus bas/le plus haut niveau. Afficher les informations d'un pokémon (s'il est dans votre équipe) ainsi qu'afficher tous les pokémon composant votre équipe. Le dossier du défi contient un fichier vous expliquant ce qu'est une liste chaînée circulaire. De plus, jeter un coup d'oeil au code fourni pour avoir une idée des fonctions que vous aurez à implémenter.

Félicitations, vous êtes maintenant prêt à débiter votre carrière de dresseur de pokémon. Will you be the very best?