

INF 731 - Travail pratique no. 1

Énoncé du problème

Un organisme de charité vous demande de concevoir une application qui permettra de faire un rapport sur les dons reçus des donateurs au fil des ans, afin d'avoir un portrait plus clair de la stratégie à adopter pour solliciter les donateurs lors de la prochaine campagne de financement.

L'organisme a cumulé, au fil des ans, un fichier qui contient des données brutes. On y retrouve, sur une ligne, un numéro de donateur qui a été assigné par l'organisme, l'année du don ainsi que le montant de ce don. Le programme que vous écrirez ne saura pas, au point de départ, combien il y a de lignes dans le fichier. Votre programme devra donc faire le nécessaire pour réserver l'espace requis pour représenter les différents dons durant le cours de l'exécution.

Les données à traiter seront inscrites dans un fichier en mode texte. Votre programme devra lire le fichier en entier, organiser les données et les traiter en fonction des résultats attendus puis enfin produire un rapport dans un fichier de texte.

Réalisation du travail

Le programme que vous devez réaliser doit être écrit en Microsoft Visual C#¹ en utilisant au mieux les concepts de la programmation orientée objet que nous avons vus et verrons dans le courant des prochaines semaines. Ce travail fait appel à de nombreuses notions que nous voyons durant les huit premières rencontres du cours – création d'une classe, création d'instances, propriétés, construction, composition, création de tableaux ou utilisation d'une collection, gestion des exceptions, méthode de classe, etc.

Il y a aussi la question de la lecture et de l'écriture dans des fichiers, mais cette difficulté est somme toute assez minime et une courte explication en classe assortie d'un exemple ou deux suffira à vous informer de la manière de procéder.

Format des données

Les données qui décriront sur une ligne le don qui a été fait par un donateur seront :

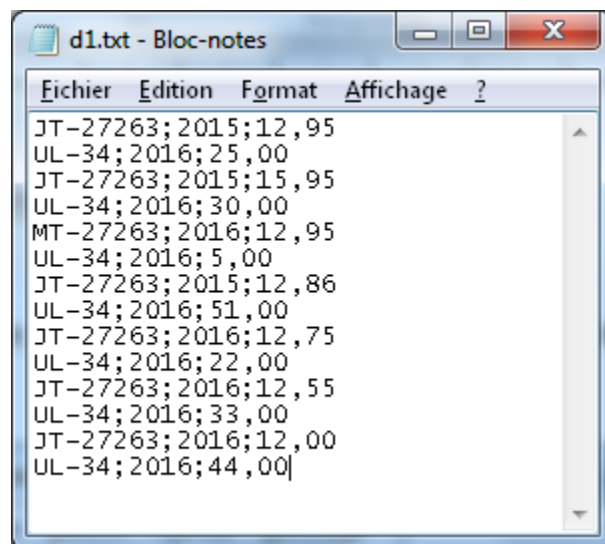
¹ Version 2013 ou 2015

- Un numéro de donateur, qui a été assigné par l'organisme. Il s'agira d'une chaîne de caractères sans espace (par exemple, on pourrait avoir la chaîne A-12345 qui a été assignée comme numéro de donateur, ou encore 123-789 par exemple); il n'y a pas de règle particulière qui s'applique au numéro de donateur outre l'absence d'espace;
- L'année du don, un entier de 4 chiffres. L'organisme ayant été créé en 1996, aucune année ne peut être plus petite que cette valeur. Puisque nous sommes en 2017, il ne sera pas possible non plus de trouver une année plus grande que 2017²;
- Le montant du don, qui doit être strictement positif, c'est-à-dire plus grand que 0.

Un fichier contient l'ensemble des dons faits au fil du temps par nos donateurs. Chaque ligne du fichier décrit un don et comporte tous les éléments mentionnés ci-haut. Vous avez l'assurance que toutes les informations sont présentes à chaque ligne. Le format d'une ligne décrivant un don est le suivant :

numéro de donateur;année;montant

Par exemple, on pourrait avoir le fichier suivant³ :



² Nous ne traiterons pas le cas d'engagements futurs à faire un don.

³ Il n'est pas exclu qu'un même donateur ait fait plusieurs dons une année donnée, ce qui explique les doublons.

Vous avez l'assurance que dans le fichier des donateurs,

1. Le numéro du donateur ne comporte pas d'espace; il n'y a pas de règle de validation du numéro;
2. Vous avez l'assurance que le deuxième élément de la ligne, qui correspond à l'année du don, est un entier correctement formé qui pourra être reconstitué sans problème. Par contre, vous devez valider que l'année se trouve dans les bornes déjà mentionnées;
3. Vous avez l'assurance que le troisième élément de la ligne, qui correspond au montant du don, est un réel correctement formé qui pourra être reconstitué sans problème. Toutefois, vous devez valider cette donnée pour vous assurer qu'elle est supérieure à 0⁴.

Dans le fichier, les éléments sont séparés les uns des autres sur la ligne par un point-virgule.

Traitement à effectuer sur les données

L'organisme vous demande d'effectuer sur les données le traitement suivant :

- On veut obtenir un rapport de l'ensemble des contributions, toute année confondue, triées par ordre d'importance en allant de la plus grande contribution à la plus petite.
- L'organisme veut obtenir aussi un rapport de la contribution totale de chaque donateur au fil des ans. On veut que ce rapport soit trié en ordre inverse du montant des dons, c'est-à-dire que le donateur dont la contribution totale a été la plus généreuse doit apparaître au début du rapport et le donateur le moins généreux doit apparaître en fin de rapport.
- Enfin, l'organisme aimerait aussi beaucoup obtenir si possible un rapport du total des contributions pour chaque année où on a des données. On veut avoir le total pour l'année la plus récente en début de rapport et le total pour l'année la plus lointaine en fin de rapport.

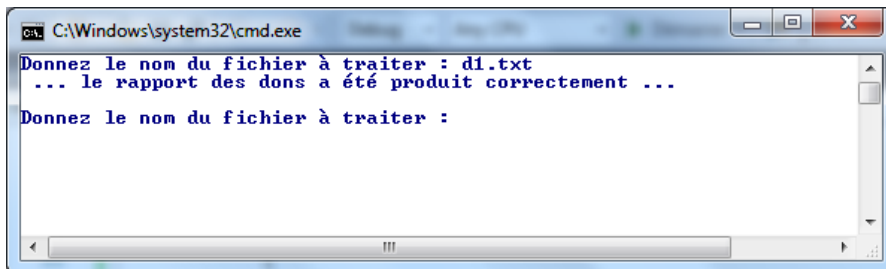
Les trois volets du rapport produit par le traitement des données doivent être écrits dans un fichier de sortie que votre programme doit créer. Ce fichier doit commencer par un entête comportant les noms des équipiers ainsi que le nom du fichier des dons lu par le programme. Par exemple, si le fichier d'entrée comprenant les dons se nomme 'd1.txt', l'entête sera :

⁴ Nous prendrons pour acquis que le montant est nécessairement une valeur qui n'a pas plus de deux chiffres décimaux.

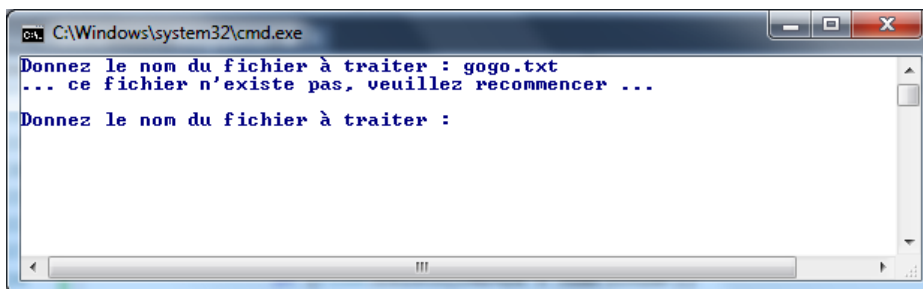
Rapport du traitement de d1.txt

par Jean-François Lefort et Lucette Labonté

Lorsque votre programme se mettra en marche, il demandera à l'utilisateur le nom du fichier d'entrée contenant la liste des dons. Il est convenu que ce fichier d'entrée se trouvera dans le dossier où résident tous les fichiers .cs du projet. Par exemple :



Si le fichier dont l'utilisateur entre le nom n'existe pas, le programme le signale à l'utilisateur et redemande le nom du fichier. Exemple :



Si le fichier existe, le programme fait la création d'un fichier de sortie qui contiendra les rapports à produire. Le fichier de sortie que votre programme doit créer portera le nom « Rapports - » suivi du nom du fichier des dons qui doit être traité. Pour le fichier d'entrée 'd1.txt', le fichier de sortie serait

'Rapports – d1.txt'

Encore une fois, il est convenu que ce fichier des résultats sera inscrit dans le même dossier que les fichiers .cs du projet.

Après avoir produit les rapports pour un fichier de données, votre programme doit demander à l'utilisateur le nom d'un nouveau fichier de dons à traiter. Lorsque l'utilisateur fait « enter » sans entrer de nom, le programme prend fin.

```

C:\Windows\system32\cmd.exe
Donnez le nom du fichier à traiter : gogo.txt
... ce fichier n'existe pas, veuillez recommencer ...

Donnez le nom du fichier à traiter : d1.txt
... le rapport des dons a été produit correctement ...

Donnez le nom du fichier à traiter :
Au revoir !
Appuyez sur une touche pour continuer...

```

Contenu du fichier de sortie – le rapport

Voici un exemple de fichier de sortie :

```

Rapport du traitement de d1.txt
par Pierre Prud'homme

Rapport pour l'ensemble des donateurs

UL-34      2016      51,00
UL-34      2016      44,00
UL-34      2016      33,00
UL-34      2016      30,00
UL-34      2016      25,00
UL-34      2016      22,00
JT-27263   2015      15,95
JT-27263   2015      12,95
MT-27263   2016      12,95
JT-27263   2015      12,86
JT-27263   2016      12,75
JT-27263   2016      12,55
JT-27263   2016      12,00
UL-34      2016       5,00

Cumulatif pour chaque donateur

UL-34      210,00
JT-27263    79,06
MT-27263    12,95

Cumulatif pour chaque année

2016      260,25
2015       41,76

```

Notez que la qualité de la présentation dans le fichier de sortie est importante et doit contribuer à bien comprendre les résultats. Il serait sans doute avisé de vous inspirer du mode de présentation que je vous suggère ici; vous pouvez bien entendu faire mieux !

Par contre, avant de vous soucier de la qualité de la présentation, vous devrez vous assurer que le traitement est correct et les résultats valables. Un résultat erroné mais bien présenté ne vaut ... pas grand chose.

Caractéristiques de la classe Don

Vous **devez** créer dans votre projet une classe `Don` pour décrire un don qui a été fait à l'organisme. Chaque instance de `Don` aura comme attributs le numéro du donateur (un string), l'année où a été fait ce don (un entier) et le montant du don en question (un float suffira amplement).

Sans égard aux services utilisés par le programme client, la classe devra offrir⁵:

- un seul constructeur qui sera paramétrique recevant en paramètre une chaîne, un entier et un float qui représenteront respectivement le numéro du donateur, l'année du don et le montant de ce don;
- un constructeur de copie;
- des propriétés permettant de modifier les attributs et de faire connaître la valeur des attributs d'un `Don`; à vous de déterminer leur degré de visibilité correct;
- une surcharge de la méthode **ToString()** afin de simplifier la production du premier rapport.

Autres consignes relatives à la conception et à l'implantation

1. Il est convenu que nous ne savons d'aucune manière combien de dons auront eu lieu au fil des années. On s'entend que leur nombre sera raisonnable⁶ sinon on aurait utilisé les services d'une base de données pour les conserver. Un tableau dont vous augmenterez la taille au besoin ou un dispositif comparable déjà disponible dans l'environnement de C# sera donc à utiliser dans le cadre de ce programme.
2. Au début de la période de développement, vous pouvez utiliser simplement un tableau de très grande dimension pour simplifier votre programme; plus tard, il sera possible de modifier cette implantation puisque la classe encapsule ses attributs.

⁵ On ne développe pas une classe juste pour les besoins du moment, bien que ceux-ci orientent nécessairement notre réflexion en période de design ou de développement.

⁶ C'est-à-dire que l'espace mémoire disponible pour un tableau sera suffisant pour contenir l'ensemble des dons.

3. Il serait sans doute approprié de conserver la liste des dons dans une classe où elle serait encapsulée; elle serait ainsi accompagnée dans la classe des méthodes permettant de traiter ces données.
4. Pensez aux validations des données : vous devriez identifier dès le début les cas à rejeter.
5. Procédez étape par étape pour faire avancer votre projet. Vous pouvez fort bien écrire une partie de la classe pour la tester avant de continuer.
6. La lecture des données devra se faire par le biais d'un objet de classe ***StreamReader***. Des explications vous seront données en classe à ce sujet.
7. L'écriture de vos résultats devra se faire par le biais d'un objet de classe ***StreamWriter***. Explications à venir également.

Critères de correction

Ce travail est corrigé sur 100 mais pondéré ensuite sur 12 points dans le calcul de la note finale du cours. **Vous devez le faire par équipe de deux personnes**⁷. Le barème de correction appliqué à ce programme sera :

- 50% des points pour le bon fonctionnement lors des tests; il y aura plusieurs tests effectués et leur pondération sera variable.
- 50% des points pour les autres critères
 - commentaires
 - respect des demandes indiquées dans cet énoncé
 - bon découpage du problème
 - bonne utilisation des concepts de la POO
 - qualité de la programmation
 - respect des normes usuelles appliquées dans le cours
 - usage de constantes lorsque la situation s'y prête
 - élégance du code
 - etc.

⁷ Au moment d'écrire ce travail, nous semblons être un nombre impair d'étudiants dans le cours. Si cela se confirme, une équipe de trois personnes devra être formée.

Les commentaires devront au moins inclure :

- Un prologue au programme indiquant
 - le nom des auteurs
 - le nom du fichier
 - le nom du projet
 - la date de création
 - la description de ce que fait ce programme
- Un prologue à chaque méthode indiquant
 - la description de ce que fait ce sous-programme
 - une description des intrants
 - une description des extrants et valeur de retour
- Chaque partie de code dont la seule lecture n'indique pas clairement la raison d'être devrait également porter un commentaire

Vous devez, lors de la remise, me fournir votre projet en entier accompagné d'un fichier **Dons.txt** qui vous a permis de tester le bon fonctionnement de votre programme. J'utiliserai ce fichier pour effectuer le premier test de vérification. Ainsi, vous êtes assuré d'avoir un certain nombre de points dans le fonctionnement de votre programme⁸. Bien entendu, je mettrai au point d'autres fichiers de tests pour valider le bon fonctionnement de vos programmes.

Remise

Vous devez remettre une version papier et une version électronique de votre programme. La version papier devra être imprimée en **mode paysage**. Le tout doit naturellement être broché. Le premier fichier qui se trouve dans la liste que vous me remettez est le fichier contenant la classe **Program** dont le **Main** sert de point d'entrée au projet. Ce fichier sera ensuite suivi de la liste des autres classes, dans l'ordre qui vous semblera le plus logique.

La date de remise est fixée à la fin de la dixième séance, soit le

Mardi 14 mars 2017 à la fin de la séance (21h45) au plus tard

Vous pourrez remettre la version électronique de votre travail en me le faisant parvenir en pièce jointe à un message de courriel à l'adresse

⁸ L'étudiant avisé s'assurera qu'il me remet un fichier **dons.txt** pour lequel son programme fonctionne correctement, ce qui l'assure d'un certain nombre de points pour le fonctionnement.

p.prudhomme@gmail.com, la date et l'heure du message faisant foi du moment de la remise. Prévoyez quelques minutes pour l'acheminement du message.

Alternativement, vous pouvez remettre la version électronique de votre travail en me donnant le soir de la remise un CD gravé contenant votre projet complet ainsi que les fichiers supplémentaires demandés. Si vous procédez ainsi, le nom de dossier contenant votre projet sur le CD doit répondre à l'exigence mentionnée au prochain paragraphe.

IMPORTANT : pour me rendre la gestion de tout cela plus facile, je vous demande de nommer votre dossier contenant le projet aux noms et prénoms des équipiers avant de le zipper; ainsi, le travail de Luc Robitaille et Wayne Gretzky me serait expédié en pièce jointe sous le nom RobitailleLuc_GretzkyWayne.zi~. N'oubliez pas non plus de supprimer les dossiers **Bin** et **Obj** avant de produire l'archive.

Bon travail.