

## IFT1169 – TRAVAIL PRATIQUE #1 – 03 mai 2006

« En avant, mettons la première! »

Mohamed Lokbani

**Équipes:** Le travail est à faire en monôme (**une seule personne**).

**Remise :** Une seule remise à effectuer : électronique le jeudi **11 mai 2006, 23h59 au plus tard, sans possibilités de retard**.

**Conseils:** N'attendez pas le dernier jour avant la remise pour commencer, vous n'aurez pas le temps nécessaire pour le faire.

**But :** Ce TP a pour but de vous familiariser avec votre environnement de travail d'une part et, d'autre part de manipuler des gadgets de base nécessaires à un programmeur en C++. Nous allons examiner en particulier les différentes options fournies sur une ligne de commande. Le but de ce travail est de vous faire pratiquer aussi la procédure de remise, la rédaction d'un rapport ainsi que le respect des échéances.

**Énoncé :** Écrire un programme C++ qui, en fonction de ses arguments, va réaliser ce qui suit :

Option	Fonction	Ligne de commande
-s	Cette option fait en sorte que le programme génère puis affiche en sortie 50 entiers compris entre une valeur min et une valeur max. « 50 » est une valeur par défaut qui peut-être modifiée avec l'option « -d ».	tp1.exe -s 10 20 Génère 50 entiers entre 10 et 20 inclusivement
-d	Cette option permet donc de changer la valeur par défaut.	tp1.exe -d 30 -s 10 20 Génère 30 entiers entre 10 et 20 inclusivement
-t	Cette option fait en sorte que le programme inverse l'ordre de ses arguments, ainsi que l'ordre des caractères dans chaque argument et affiche le résultat en sortie.	tp1.exe -t rien a signaler Affichage : relangis a neir
-h	Cette option fait en sorte que le programme affiche en sortie un court message expliquant comment il doit être utilisé.	tp1.exe -h
	Idem que « -h »	tp1.exe

**Mixage des options :** On vous demande de ne pas traiter les cas suivants :

tp1.exe -s -t -h (ou -s -d -t -h)
tp1.exe -s -t (ou -s -d -h)
tp1.exe -t -h

Ces exemples signifient que les options principales (« -s », « -t » et « -h ») ne seront jamais disponibles en même temps sur la même ligne de commande.

**Erreurs à traiter :**

L'option « -s » accepte forcément deux entiers (ni plus ni moins) comme arguments. Sinon, il y a erreur.

L'option « -d » est donc optionnelle. Si elle est présente sur la ligne de commande, l'option « -s » doit être aussi présente. Sinon, il y a erreur.

L'option « -t » accepte un mot et plus. Sinon, il y a erreur. Un mot peut-être constitué d'une seule lettre (comme le montre l'exemple « rien a signaler »).

L'option « -h » n'accepte aucun argument.

### **Bonus :**

En bonus, on vous demande d'oublier le paragraphe « Mixage des options » et de traiter dans ce cas toutes les options possibles qui peuvent se présenter sur la ligne de commande. À noter que si l'option « -h » est présente en même temps qu'une autre option, elle est prédominante sur les autres options. De ce fait, les autres options ne seront donc pas exécutées.

tp1.exe -h -s 10 20	Équivalent à « tp1.exe -h »
tp1.exe -h -d	Normalement c'est une erreur, car l'option « -d » doit être toujours accompagnée d'une valeur en plus de l'option « -s ». Mais comme l'option « -h » a prédominance sur les autres options, il n'y aura pas d'erreur. Équivalent à « tp1.exe -h ».

### **Quelques informations utiles :**

- Pour générer des nombres aléatoires en « C++ », utiliser la fonction « rand » définie dans le fichier d'entête « cstdlib ».
- Pour manipuler les options présentes sur une ligne de commande, utiliser les fonctionnalités définies dans « getopt.h ». Voir aussi les notes de cours « chapitre 2 ».

### **Hypothèses et contraintes :**

- Pour commencer, vous devez déjà respecter les différentes contraintes de programmation précédemment décrites.
- C'est un travail à faire **SEUL** ! Et seul ne signifie pas deux ni dix. Seul signifie aussi **réfléchir seul**. Plagiat équivaut à un 0 pour commencer.
- IFT1169 est un cours avancé en C++, donc votre programme doit être écrit en C++ et pas en C ni en Java! Nous n'autoriserons aucune référence au langage C. Par exemple l'instruction [#include <stdio.h>] fait référence au langage C, donc elle n'est pas autorisée. Utiliser plutôt [#include <iostream>].
- Ce travail n'est pas un exercice d'algorithmique, donc pas besoin de compliquer le travail pour rien!
- Vous vous assurez de ne pas changer le nom de fichier. Vous devez respecter aussi le format de l'affichage en sortie.

**Remise :** Il est important de noter que votre TP sera compilé avec gcc3.2.4. Si par choix vous décidez d'utiliser un autre compilateur, vérifiez que le code que vous avez produit (qui normalement fonctionne correctement chez vous) fonctionne bien sur les ordinateurs de la DESI. Pour avoir la version du compilateur, utilisez la commande "**gcc -v**", qui devra donner le numéro de version "**3.2.4**".

Par ailleurs, assurez-vous de la présence de l'option « pedantic » sur la ligne de compilation. (Cette option n'est pas activée par défaut dans l'utilitaire « devcpp ». Il faudra donc penser à l'activer).

Vous devez remettre 2 fichiers : « tp1.cpp » et « rapport ». Le fichier « tp1.cpp » va contenir votre code alors que le fichier « rapport » va décrire comment vous avez procédé pour réaliser ce travail (pour le contenu d'un rapport voir la FAQ sur la page web du cours). Vous devez prévoir deux sortes de rapport : un guide d'utilisation de votre programme qui sera lu par un utilisateur néophyte et un guide pour programmeur lui expliquant la démarche suivie pour la réalisation de ce travail.

Commencer d'abord par vous connecter sur la machine « remise » comme suit : « ssh remise ». Par la suite :

1. Envoyer vos fichiers « tp1.cpp » et le rapport par la procédure de remise électronique habituelle (Pour obtenir de l'aide sur cette commande, tapez dans un Xterm : man remise). Respecter les noms des fichiers.

**remise ift1169 tp1 tp1.cpp rapport**

2. Vérifier que la remise s'est effectuée correctement.

**remise -v ift1169 tp1**

**Barème :** Ce TP1 est noté sur 7 points (+2 points pour le bonus).

<b>Compilation et respect des spécifications</b>	<b>0.5</b>
<b>Codage, commentaires etc.</b>	<b>2</b>
<b>Rapport</b>	<b>1.5</b>
<b>Tests (fournis)</b>	<b>3</b>
<b>Bonus</b>	<b>2</b>

**En plus du précédent barème, vous risquez de perdre des points dans les cas suivants ....**

- La non remise électronique (volontaire ou par erreur) est sanctionnée par la note 0.
- La non remise papier vous pénalise de 1 point.
- Les programmes ne contenant pas d'en-tête, -1 point.
- Un programme qui ne compile pas : 0.
- Un programme qui compile mais ne fait pas les choses prévues dans la spécification : 0.
- Les avertissements (warnings) non corrigés : cela dépend de la quantité! À partir de -0.25 et plus.
- Le non respect du nom du fichier entraîne une erreur de compilation donc un des points de la spécification n'a pas été respecté : 0.
- Aberration dans le codage : Même si tous les chemins mènent à Rome, faites l'effort nécessaire pour éviter de prendre le plus long!

**Automatisation de la correction :** Nous allons utiliser des scripts pour compiler et exécuter votre programme. Assurez-vous de respecter le nom du fichier « tp1.cpp », ainsi que les noms des différentes options.

Ainsi donc, au moment de la remise de votre travail, vous avez déjà une idée approximative de la note que vous allez obtenir.

### **Des questions à propos de ce TP?**

Une seule adresse : [dift1169@iro.umontreal.ca](mailto:dift1169@iro.umontreal.ca)

Pour faciliter le traitement de votre requête, inclure dans le sujet de votre email, au moins la chaîne: [IFT1169] et une référence au **tp01**.

### **Mise à jour**

03-05-2006 diffusion de l'énoncé.