

TD 1 : premiers pas en C++

Jérémy Foulon

17 janvier 2012

Compiler et exécuter un programme C++

Introduction à C++ : <http://www.yolinux.com/TUTORIALS/LinuxTutorialC++.html>.

Installation de Cygwin sur windows

Documentation : <http://www.ann.jussieu.fr/~hecht/ftp/cygwin/>.

Éditeur de texte

XEmacs : <http://www.xemacs.org/>.

Compilation du programme *Hello World*

file : <http://www.ann.jussieu.fr/hecht/ftp/InfoSci/s1/HelloWord.cpp>.

compilation : `g++ -Wall -o main HelloWorld.cpp` . Création d'un exécutable du programme portant le nom : `main`. execution : `./main`.

L'option `-Wall` demande au compilateur d'afficher un maximum de "warnings" qu'il faut s'assurer d'éliminer.

Entrées/Sorties

Saisie et affichage dans le terminal

Utilisation des fonctions `cin` et `cout`. Pour utiliser les entrées/sorties, on doit inclure le fichier d'en tête : `<iostream>`. Il faut ajouter au début du fichier la ligne suivante : `# include <iostream>`. Par commodité on se place dans le namespace standard (`using namespace std;`).

Exercice 1

- Écrire un programme permettant à l'utilisateur de saisir deux entiers et d'afficher leur somme.
- Faire la somme de deux entiers positifs en indiquant une saisie erronée à l'utilisateur si lors de la saisie l'entier est négatif.
- Ajouter deux fonctions qui respectivement déterminent l'entier le plus grand et l'entier le plus petit.

Exercice 2

Rappel : tableau en C++, *type nom_du_tableau[nombre_éléments]* ;

- Écrire un programme permettant à l'utilisateur de saisir autant de valeurs qu'il le souhaite et le stocker dans un tableau.
- Afficher l'ensemble des valeurs saisies.

Saisie et affichage dans un fichier

Inclure l'en tête : `# include <fstream>`.

Exercice 3

- Créer un fichier *input.dat* contenant le nombre de valeurs à lire sur la première ligne et sur la ligne suivante toutes les valeurs.
- Écrire à l'aide de la fonction *ifstream* lire le contenu du fichier.
- Réécrire ce fichier en multipliant toutes les valeurs par 10 avec la fonction *ofstream* dans le fichier *output.dat*.

Exercice 3 bis

- Donner les noms de fichier à lire et à écrire en argument à l'exécution du programme.

Utilisation de Gnuplot

Installation

Téléchargement : http://www.ann.jussieu.fr/hecht/ftp/InfoSci/doc-html/alire_cygwin.html

Première utilisation

Visualisation de la fonction $f(x)=\cos(x)$, $x=[0,\pi]$; les valeurs sont affichées sur deux colonnes correspondant à x et $f(x)$. Définition d'une fonction :

Exercice 4

- Définir une discrétisation (découpage de l'intervalle x en n sous-intervalles).
- Évaluer la fonction cosinus sur chaque $x[i]$.
- Écrire dans le fichier "résultat.dat" le fichier gnuplot associé à la visualisation de cosinus.
- Visualiser avec gnuplot : `plot "resultat.dat" w l`.

Divers

Exercice 5

Un manutentionnaire est payé 4,75 Euros/h et travaille 8 heures par jours. Il doit soulever 1200 colis par jours et reçoit 4,50 Euros/100 colis supplémentaires (pas de pénalités s'il soulève dans la journée moins de 1200 colis).

- Écrire un programme qui permet à un manutentionnaire de calculer rapidement son salaire journalier.

Exercice 6

Écrire un programme qui lit la largeur et la longueur d'un rectangle et affiche :

- un rectangle plein d'étoiles (utiliser le caractère *)
- le bord du rectangle uniquement (utiliser le caractère *)