

TD/TP - Programmation C

1 Prise en main de l'environnement - machine virtuelle Linux

Ouvrir un terminal, se déplacer dans l'arborescence des fichiers (`cd`, `pwd`, `ls`), copier un fichier `cp`, le déplacer `mv`, créer un répertoire `mkdir`

2 Compilation

On écrit un programme dans un fichier source. Il est de coutume de donner un nom qui porte l'extension `.c` aux fichiers qui contiennent des programmes en langage C. On sauvegarde donc un programme édité ci-dessus par n'importe quel éditeur de texte dans un fichier appelé `programme.c`. Le contenu de ce fichier est compréhensible par le programmeur, mais pas par la machine. Il faut transformer le code pour qu'il soit compréhensible par l'ordinateur. Pour cela, on utilise un compilateur. Il s'agit d'un programme qui va interpréter le fichier source écrit par le programmeur et créer un fichier exécutable. Nous utilisons le compilateur `gcc` (GNU Compiler Collection). On compile le programme par la commande : `gcc -o programme programme.c` (Attention, c'est le nom de l'exécutable qui doit suivre `-o`, pas le nom du fichier source). Une fois le fichier exécutable `programme` généré, on lance le programme par la commande `./programme`.

3 Echauffement

1. Premier programme Créer, avec `nedit` ou l'éditeur de texte de votre choix, un fichier `hello.c` qui contient les instructions suivantes :

```
#include <stdio.h>
main()
{
    printf ("Hello\n");
}
```

Le compiler par la commande `gcc -o hello hello.c`. Lancer son exécution par `./hello`.

2. Entrées/Sorties Écrire un programme C qui demande à l'utilisateur d'entrer un entier et un réel et qui affiche ensuite ces deux nombres. Le compiler et l'exécuter.

3. Carré Écrire un programme C qui demande à l'utilisateur la longueur du côté d'un carré, et qui affiche le périmètre, la surface et la longueur de la diagonale de celui-ci (la fonction `sqrt` se trouve dans la librairie `math.h`, il faut alors compiler le programme avec l'option `-lm`).

4. Mois Écrire un programme qui affiche le nombre de jours dans le mois donné par l'utilisateur (l'utilisateur entrera un nombre de 1 à 12).

5. Voyage Écrire un programme qui calcule la durée d'un trajet connaissant l'heure de départ et d'arrivée. On se contente des heures et des minutes, la durée totale ne dépassera jamais 24 heures.

6. Lendemain Écrire un programme qui affiche le lendemain d'une journée donnée (jour, mois, année). On ne tiendra pas compte ici des années bissextiles, le mois de février aura toujours 28 jours. Si vous avez fini, améliorez le programme pour qu'il tienne compte des années bissextiles.

4 Boucles

1. Sommes Écrire un programme C qui lit un entier n , calcule une des sommes suivantes et affiche le résultat :

1. $1 * 1 + 2 * 2 + \dots + n * n$
2. $1 + 3 + \dots + 2 * n - 1$
3. $2 + 4 + \dots + 2 * n$
4. $1 + (1 + 2) + \dots + (1 + 2 + \dots + i) + \dots + (1 + 2 + \dots + n)$
5. $1 + 1/2 + 1/3 + \dots + 1/n$

2. Factorielle Écrire un programme C qui calcule la factorielle $n!$ d'un nombre n lu au clavier, tant que l'utilisateur le demande. Rappel : $n! = 1 * 2 * \dots * n$ et $0! = 1$.

3. Nombres premiers Écrire un programme C qui affiche tous les nombres premiers de 1 à 100. Rappel : Un nombre est premier s'il n'est divisible que par 1 et par lui-même.

4. Suites Écrire un programme C qui lit un entier n et affiche le n -eme élément de la suite suivante :

- a) $u_1 = 1$; $u_n = 2 * u_{n-1} + 3$
b) $u_1 = 1$; $u_2 = 1$; $u_n = u_{n-1} + u_{n-2}$ (suite Fibonacci)

5. Table de multiplication Écrire un programme C qui affiche la table de multiplication d'un entier n lu au clavier. Écrire ensuite un programme C qui affiche la table des produits pour n variant de 1 à 10.

6. Compter les voyelles Écrire un programme C qui calcule le nombre de voyelles (ou de consonnes) dans un texte non accentué et terminé par un caractère $\#$. On suppose que le texte n'est composé que de lettres minuscules non accentuées, que les mots ne sont séparés que par des espaces, qu'il n'y a aucune ponctuation et qu'il n'y a pas de caractère retour-chariot.

77. Deviner un nombre Écrire un programme C qui tire au sort un nombre de 1 à 100 et demande à l'utilisateur de le deviner. Si l'utilisateur a trouvé le nombre, alors le jeu s'arrête ; sinon l'ordinateur répond *plus petit* ou *plus grand*. L'utilisateur a le droit à cinq tentatives de deviner le nombre.

8. Deviner un nombre 2 Écrire un programme C qui tire au sort un nombre de 1 à 100 et demande à l'utilisateur de le deviner. Si l'utilisateur a trouvé le nombre, alors le jeu s'arrête ; si l'écart entre le nombre recherché et la réponse de l'utilisateur est

- plus petit que 30, alors l'ordinateur répond *très froid* ;
- plus petit que 20, alors l'ordinateur répond *froid* ;
- plus petit que 10, alors l'ordinateur répond *?* ;
- plus petit que 5, alors l'ordinateur répond *chaud* ;
- plus petit que 2, alors l'ordinateur répond *très chaud*.

L'utilisateur a le droit à cinq tentatives de deviner le nombre.

9. Le jeu des allumettes Chaque joueur puise à tour de rôle de 1 à p allumettes dans un tas comprenant initialement N allumettes. Celui qui prend la dernière allumette gagne le jeu. Le programme vérifie seulement la validité des choix des 2 joueurs et désigne le vainqueur.

10. triplets Écrire un programme C qui affiche les triplets d'entiers (i, j, k) , tel que $j^2 = i^2 + k^2$ (n donné par l'utilisateur) et $i^2 + j^2 = k^2$. Modifier le programme pour que (i, j, k) vérifie $i^3 + j^3 = k^3$.

11. Puissance Calculer x puissance n , pour x et n entiers lu au clavier.

12. Rectangle Écrire un programme C qui lit la largeur et la longueur d'un rectangle et affiche :

1. un rectangle plein d'étoiles ;
2. le contour du rectangle.

13. Triangle Écrire un programme C qui lit une des cotes d'un triangle et affiche :

1. un triangle plein d'étoiles ;
2. le contour du triangle.

14. Rebours Écrire un programme C qui lit un nombre positif entré au clavier comme une suite de chiffres non nuls, terminée par zéro, et qui affiche le nombre entier lu à rebours. (Contrôlez s'il s'agit vraiment de chiffres). Exemple : Entrée : 1 2 3 4 0 Affichage : 4321

15. PGCD Écrire un programme C qui calcule le P.G.C.D. de deux entiers naturels entrés au clavier en utilisant l'algorithme d'Euclide.