

Séance 1: Introduction

Afin d'organiser proprement vos fichiers, créez dans votre compte un répertoire nommé *ProgC* dans lequel vous allez créer le répertoire *tp1*.

Exercice 1. Compilateur C et affichage à l'écran

Pour écrire les programmes en C, vous pouvez utiliser l'éditeur de texte *nedit* : il reconnaît le langage C et vous montrera la syntaxe en couleur, à condition que les noms des fichiers sources se terminent par *.c* !!

Pour lancer *nedit*, tapez dans le terminal la commande *nedit&*. Le *&* vous permet de garder la main dans le terminal.

1. Tapez le programme suivant dans un fichier nommé *essai.c* (veillez à soigner l'indentation !)

```
#include <stdio.h>

main()
{
    int entier;
    printf("Entrez un entier : ");
    scanf("%d",&entier);           // Saisie de l'entier au clavier
    printf("Bonjour, vous avez saisi %d !!!", entier); // Affichage à l'écran
}
```

2. Compilez le programme. Pour cela, lancez dans le terminal la commande :
gcc -o essai essai.c

Le compilateur gcc crée un fichier exécutable. Par défaut, tous les exécutables créés par gcc se nomment *a.out*. (vous pouvez le vérifier en tapant la commande *gcc essai.c*).

L'option *-o* permet de donner le nom que l'on souhaite à cet exécutable, suivant la syntaxe :
gcc -o NomFichierExécutable NomFichier.c

Avec la commande *ls -l*, regardez le fichier *essai.out* qui a été créé : vous voyez dans ses droits d'accès qu'il est exécutable.

3. Pour exécuter *essai.out*, tapez dans le terminal la commande *./essai*
Le *./* indique à Unix que le fichier à lancer se trouve dans le répertoire courant.
4. Pour passer à la ligne à la fin de l'affichage, utilisez la séquence d'échappement *\n* (les mêmes séquences d'échappement qu'en Java sont disponibles : *\t, \n, \'*).
Que se passe-t-il si vous saisissez un nombre réel au clavier ? Et si vous saisissez un caractère ? Pourquoi ?
5. Regardez ce qui se passe à la compilation si vous oubliez la parenthèse à la fin de l'instruction *printf* ou le point-virgule à la fin de la ligne d'affichage à l'écran.

Exercice 2. Formats pour l'affichage écran

1. Ecrire dans un fichier *affichage.c* un programme qui lit au clavier un caractère (char), et affiche ensuite ce caractère (avec le format %c) et son code ASCII (avec le format %d).
2. Modifier le programme précédent afin de lire au clavier un float et de l'afficher avec %f et %d. Que se passe-t-il ? Peut-on avoir un affichage juste ? Que se passe-t-il si vous déclarez un entier, et que vous essayez de lire sa valeur au clavier avec le format %f ? Pourquoi ?

Exercice 3. Moyenne

Ecrire dans un fichier *moyenne.c* un programme qui lit au clavier des entiers positifs et qui affiche à la fin le nombre d'entiers saisis par l'utilisateur ainsi que leur moyenne. La saisie s'arrête lorsque l'utilisateur entre une valeur négative.

Exercice 4. Boucle

Ecrire un programme *table.c* qui saisit un entier valeur au clavier et qui affiche la table de multiplication jusqu'à l'entier *valeur* de tous les entiers allant de 1 jusqu'à valeur sous la forme suivante (avec valeur = 5):

1	2	3	4	5
2	4	6	8	10
3	6	9	12	15
4	8	12	16	20
5	10	15	20	25

Utilisez les tabulations (\t) pour obtenir un affichage correct en colonnes.