# Rappels (C)

## Avantages du langage:

- Efficace (registres, pointeur, opérations sur les bits, pas de contrôle à l'exécution, etc...)
- Grande liberté du programmeur
- Bibliothèque très étendue
- Interface claire avec Unix (et indépendant de l'OS)
- Portable

#### **Inconvénients:**

- Syntaxe à deux niveaux (préprocesseur)
- Responsabilité du programmeur (à cause de la grande liberté qui lui est donnée)
- Erreurs de compilation peut explicites
- Langage ancien (pas d'objets, de généricité, etc...)

## Rappel de la structure :

```
#include <stdio.h>
int main() {
    int result = 0;

    scanf("%d", &result);
    if(result < 0) {
        result = -result;
    }
    printf("%d \n", result);
}</pre>
```

## Directive de compilation

→ Quand précédé de #, parle au préprocesseur Ici la valeur est remplacée par la suite, mais il toujours préférable d'initialiser les variables.

Lie un entier au clavier, et le place dans l'adresse mémoire de la variable « result ».

Affiche la variable « result » et un saut de ligne.

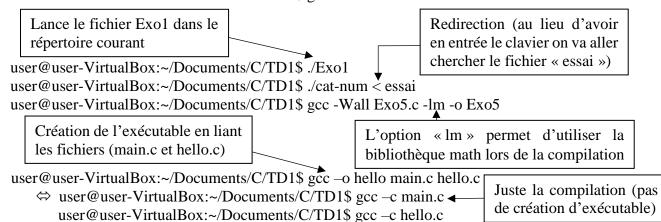
## **Quelques types:**

Type	Code
int	%d
long	%ld
float	%f
double	%lf
char	%c
string (⇔char*)	%s
pointeur (⇔ void*)	%p
short	%hd
Entier en héxa	%x

Compile le fichier Exo1.c et crée un fichier exécutable nommé « Exo1 » (Wall : Warning all)

#### **Commandes utiles:**

user@user-VirtualBox:~/Documents/C/TD1\$ gcc -Wall Exo1.c -o Exo1



user@user-VirtualBox:~/Documents/C/TD1\$ gcc —o hello main.o hello.o

Quelques exemples de petits programmes :

```
Consigne:
                                                               Code:
Écrire le programme cat-num qui affiche en
                                             #include <stdio.h>
les numérotant les lignes lues sur le fichier
                                             #include <stdbool.h>
standard d'entrées.
                                             int main() {
                                                    char c:
                                                                          EndOfFile
                                                    bool newL = false;
                                                                          (dernier caractère)
                                                    int ligne=1;
                                                    printf("%d. ", ligne);
                                                    while((c = getchar()) != EOF)
                                                           if(newL == true)
                                                                  printf("%d. ", ligne);
                                                                  newL = false;
                                                           printf("%c",c);
                                                           if(c=='\n')
                                                                  ligne++;
                                                                  newL = true;
                                                    return 0;
Écrire un programme qui affiche une table de
                                             #include <stdio.h>
conversion degré Celsius-degré Fahrenheit
                                             #include <math.h>
avec un pas de 0,5. Il faudra arrondir les
                                             int main() {
Fahrenheit à l'entier le plus proche.
                                                    for(float i=0.0; i<=20; i+=0.5){
                                                           printf("Celcius :
                                             %5.1lf\t\tFahrenheit:
                                             d^*(9*i)/5+32);
                                                                     Arrondit
                                                    return 0;
```