

## 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);
}
```

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 hexa	%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

Lance le fichier Exo1 dans le répertoire courant

user@user-VirtualBox:~/Documents/C/TD1\$ ./Exo1

user@user-VirtualBox:~/Documents/C/TD1\$ ./cat-num < essai

user@user-VirtualBox:~/Documents/C/TD1\$ gcc -Wall Exo5.c -lm -o Exo5

Redirection (au lieu d'avoir en entrée le clavier on va aller chercher le fichier « essai »)

Création de l'exécutable en liant les fichiers (main.c et hello.c)

L'option « lm » permet d'utiliser la bibliothèque math lors de la compilation

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

↔ user@user-VirtualBox:~/Documents/C/TD1\$ gcc -c main.c

user@user-VirtualBox:~/Documents/C/TD1\$ gcc -c hello.c

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

Juste la compilation (pas de création d'exécutable)

### Quelques exemples de petits programmes :

Consigne :	Code :
Écrire le programme <i>cat-num</i> qui affiche en les numérotant les lignes lues sur le fichier standard d'entrées.	<pre> #include &lt;stdio.h&gt; #include &lt;stdbool.h&gt;  int main() {     char c;     bool newL = false;     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; } </pre> <div data-bbox="1173 409 1436 504" style="border: 1px solid black; padding: 5px; display: inline-block;">         EndOfFile (dernier caractère)     </div>
Écrire un programme qui affiche une table de conversion degré Celsius-degré Fahrenheit avec un pas de 0,5. Il faudra arrondir les Fahrenheit à l'entier le plus proche.	<pre> #include &lt;stdio.h&gt; #include &lt;math.h&gt;  int main() {     for(float i=0.0;i&lt;=20;i+=0.5){         printf("Celcius : %5.1lf\t\tFahrenheit : %d\n",i,(int)rintf(((9*i)/5)+32));     }     return 0; } </pre> <div data-bbox="1114 1310 1268 1373" style="border: 1px solid black; padding: 5px; display: inline-block;">         Arrondit     </div>