

# TP 2 – premiers pas en langage C

## 1. Objectif

Le but de ce TP est de se familiariser avec l'environnement de programmation en langage C.

**Vous trouverez les codes source dans le fichier « src.zip » sur la plateforme foad.ensicaen.fr dans le cours « Initiation à la programmation - Elec ».**

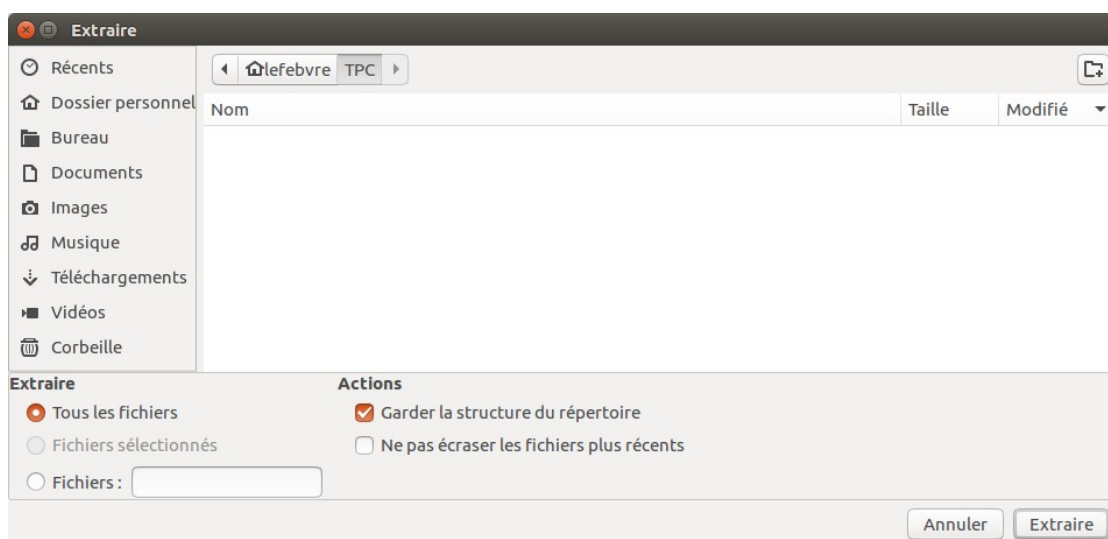
## 2. Exercices

```
#include <stdio.h>
int main(int argc, char* argv[]) {
    int promo ; // declaration de la variable promo
    printf("Donnez l'année de votre promo : ") ; // affiche un message
    scanf("%d", &promo) ; // met ce que l'utilisateur a saisi dans la variable promo
    if (promo > 0 ) {
        printf("Bienvenue à la promotion %d !\n", promo) ;
    }
    else {
        printf("Promo doit etre un nombre positif !\n") ;
    }
    return (0) ;
}
```

Pour créer le répertoire de travail TPC, écrivez dans une console (ne tapez pas le \$ et ce qui précède )

```
lefebvre@pclefebvre2:~$ cd
lefebvre@pclefebvre2:~$ mkdir TPC
lefebvre@pclefebvre2:~$ cd TPC
lefebvre@pclefebvre2:~/TPC$
```

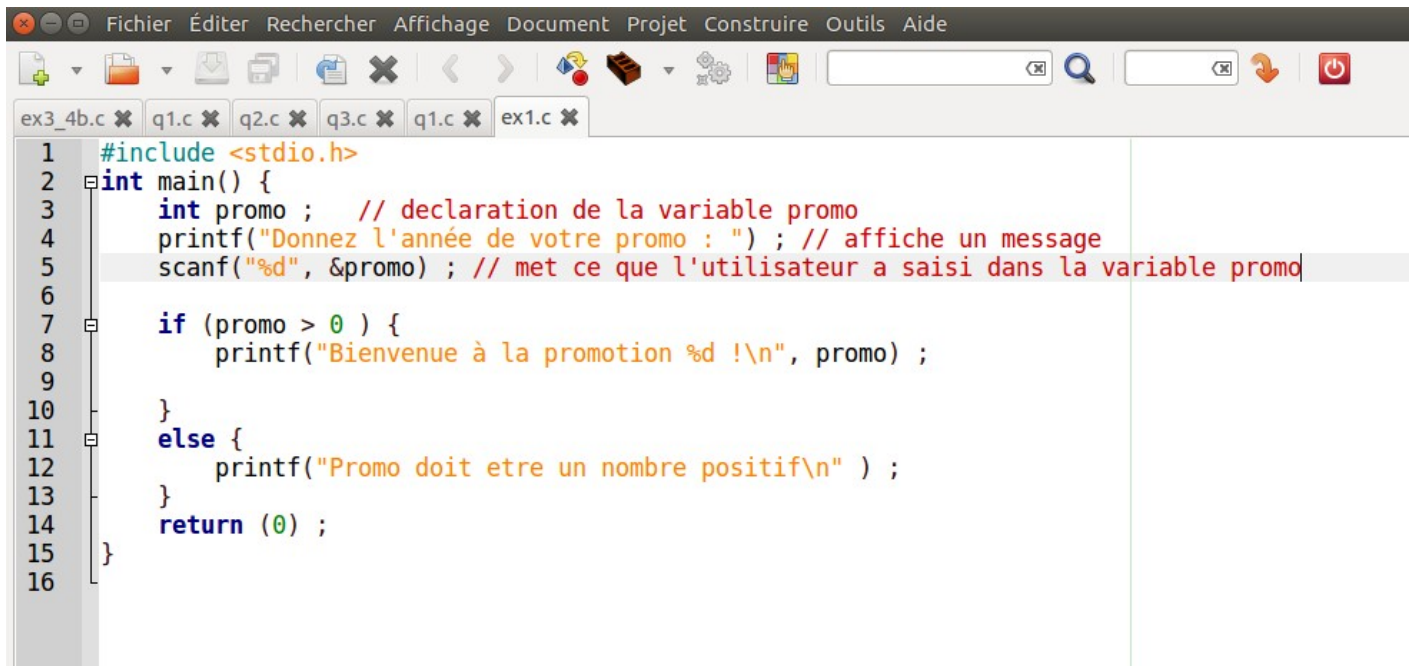
nous allons extraire l'archive **src.zip** dans le répertoire TPC. Pour cela nous allons travailler de manière classique. Après avoir téléchargé le fichier src.zip sur la plateforme, double-cliquez dessus, sélectionner le répertoire « src » et cliquer sur extraire. Retrouvez dans votre répertoire personnelle le répertoire TPC puis cliquer sur extraire pour finaliser l'extraction.



Maintenant écrivez dans la console :

```
Terminal Fichier Édition Affichage Rechercher Terminal Aide
lefebvre@pclefebvre2:~/TPC$ cd src
lefebvre@pclefebvre2:~/TPC/src$ cd tp2
lefebvre@pclefebvre2:~/TPC/src/tp2$ geany ex1.c &
[1] 24122
lefebvre@pclefebvre2:~/TPC/src/tp2$
```

Geany affiche alors le programme comme ceci :



```
Fichier Éditer Rechercher Affichage Document Projet Construire Outils Aide
ex3_4b.c x q1.c x q2.c x q3.c x q1.c x ex1.c x
1 #include <stdio.h>
2 int main() {
3     int promo ; // declaration de la variable promo
4     printf("Donnez l'année de votre promo : ") ; // affiche un message
5     scanf("%d", &promo) ; // met ce que l'utilisateur a saisi dans la variable promo
6
7     if (promo > 0 ) {
8         printf("Bienvenue à la promotion %d !\n", promo) ;
9
10    }
11    else {
12        printf("Promo doit etre un nombre positif\n" ) ;
13    }
14    return (0) ;
15 }
16
```

Retournez dans la console avec les touches [alt] [tab] et tapez ceci pour **compiler** le programme :

```
$ gcc ex1.c -o ex1.bin
```

S'il y a des erreurs corrigez-les, sinon tapez ceci pour **exécuter** le programme :

```
$ ./ex1.bin
```

*Remarques :*

*les erreurs sont affichées de cette manière :*

```
ex1.c: In function 'main':
```

```
ex1.c:3:13: error: expected '=', ',', ';', 'asm' or '__attribute__' before 'mo'  
int pro mo ; // declaration de la variable promo
```

*Cela veut dire que l'erreur est ligne 3 colonne 13 dans le fichier ex1.c*

*Attention, le compilateur affiche l'erreur là où **il commence à ne plus comprendre** ce que vous avez écrit. Il se peut que la « vraie » erreur soit bien avant dans votre programme.*

Maintenant que vous avez compris que vous devez d'abord compiler le programme avant de l'exécuter, vous pouvez utiliser les icônes de geany qui vont faciliter vos développements futurs :

- Pour compiler le programme appuyez sur l'icône « briques ».
- Pour exécuter le programme appuyez sur l'icône « engrenage ».

1) **Modifiez** le programme pour qu'il demande l'année en cours et affiche « Pas possible » si promotion est

supérieur à l'année en cours +3.

Exemple, si l'année en cours est 2042, le programme affiche « Pas possible » si l'utilisateur entre un numéro de promotion égal à 2046.

2) **Modifiez** le programme afin qu'il teste également si promotion est bien compris entre l'année en cours+1 et l'année en cours +3. (*corrigé disponible*)

3) **Écrire** un programme qui demande 3 nombres entiers et affiche la moyenne entière. On utilisera une variable somme qui sera initialisée à 0. On additionnera les nombres au fur et à mesure qu'ils sont entrés au clavier.

4) Même chose avec des nombres flottants.

5) **Écrire** un programme C qui demande 3 nombres et affiche le plus petit des 3 nombres.

Voici le programme ex2.c

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>

int main() {
    int a = 0 ;
    int n ;
    srand(time(NULL)) ; // initialise le générateur de nombre aléatoire
    do {
        n = rand() % 20 ; // tire un nombre aléatoire compris entre 0 et 19
        printf ("nombre = %d\n", n) ;
        a=a+1 ;           // incrémente a de 1
    } while (a <5) ;    // fait ce qui est enter do et while 5 fois
    return (0) ;
}
```

6) Que fait ce programme ?

7) Mettez en commentaire l'instruction `srand(time(NULL))` . Compilez le programme et exécutez le **plusieurs fois**. Que constatez-vous ?

8) *Pour les plus rapides*, écrire un programme pour réaliser le jeu classique suivant : « L'ordinateur tire un nombre entre 0 et 100, et l'utilisateur doit deviner ce nombre. A chaque essai de l'utilisateur, la machine indique s'il est plus petit ou plus grand. »