TP 3 – printf, scanf, tests et boucles

1. Objectif

Le but de ce TP est de se familiariser avec les instructions de boucle.

2. Exercice

1) Editer le fichier ex1.c que vous trouverez dans le répertoire /src/tp3 de l'archive. **Qu'affiche** ce programme dont le code est ci-dessous ?

```
/**
 * @file ex1.c
 * @author : Philippe Lefebvre <philippe.lefebvre@ensicaen.fr>
 * @date : 2020/12/25
 * @brief : Programme mystérieux…
 * @version : 1.0
**/

#include<stdio.h>
#include<stdlib.h>
int main ( void ) {
    int i,s=0;
    for (i=0; i<5; i++){
        s=s+i ;
    }
    printf ("s=%d\n", s) ;
    return (EXIT_SUCCESS) ;
}
```

2) **Remplacez** « `for (i=0; i<5; i++)` » par « `for (i=0; i<=5; i++)` ». Qu'affiche maintenant le programme ?

3)

a. Modifiez le programme pour que le nombre d'itérations (précédemment égal à 5) soit demandé à l'utilisateur.

b. Écrire un programme qui calcul la factoriel d'un nombre demandé à l'utilisateur. Le programme doit redemander un nombre si un nombre négatif a été entré.

4) **Écrire** un algorithme permettant d'imprimer le triangle suivant, le nombre de lignes étant donné par l'utilisateur. (*corrigé disponible)*

```
0
01
012
0123
01234
012345
0123456
01234567
012345678
```

5) **Qu'affiche** le programme ci-dessous (code dans l'archive) ? On remarquera au passage que `nb=nb/10` est une division entière.

```
/**
 * @file ex5.c
 * @author : Philippe Lefebvre <philippe.lefebvre@ensicaen.fr>
 * @date : 2020/12/25
 * @brief : Programme mystérieux…
 * @version : 1.0
**/

#include<stdio.h>
int main (void) {
    int nb=7531, i=0 ;
    while ( nb>0 ) {
        nb=nb/10 ;
        i=i+1 ;
    }
    printf ("nombre de chiffres = %d\n", i) ;
}
```

6) **Modifiez** le programme pour qu'il demande « nb » à l'utilisateur et qu'il affiche « Le nombre de chiffres de 7531 est 4 ».

7) **Qu'affiche** le programme ci-dessous (code dans l'archive) ?

```
/**
 * @file ex_7.c
 * @author : Philippe Lefebvre <philippe.lefebvre@ensicaen.fr>
 * @date : 2020/12/25
 * @brief : Programme mystérieux…
 * @version : 1.0
**/

#include<stdio.h>

int main (void) {
    int j=3, p=1 ;
    do {
        p=p*10 ;
        j-- ;
    } while (j>0) ;
    printf ("p = %d \n", p) ;
}
```

8) On va chercher à afficher les chiffres d'un nombre entier quelconque, par exemple le nombre 7531. Pour effectuer cela, vous devrez d'abord calculer par une boucle la valeur de $p=10^{nb_de_chiffres\ -\ 1}$. Puis vous calculerez c1=7531/ p. Ensuite vous calculerez c2=(7531-c1*p)/(p/10)... Ce qui donne l'algorithme suivant :

```
nb ← 7531, i ← 0, nb2 ← nb
tant que nb > 0 faire
    nb ← nb/10 ;
    i ← i+1 ;
fin tant que

p ← 1, j ← i
tant que j>0 faire
    p ← p*10
    j ← j-1
fin tant que

tant que p>0 faire
    c ← nb2/p
    afficher c
    nb2 ← nb2 – c*p
    p ← p/10
fin tant que
```

Écrivez le code C de cet algorithme.

9) **Modifiez** le programme pour qu'il affiche la représentation binaire de 7531. Bien avoir à l'esprit que la machine ne possède pas la représentation en base 10 du nombre, mais, dans le programme précédent, c'est bien la division successive par 10 qui a fait apparaître les chiffres en base 10.

10) L'algorithme tel qu'il est écrit ajoute un 0 devant les chiffres. Modifiez l'algorithme afin que seuls les chiffres significatifs soient affichés.

11, facultatif) **Modifiez** le programme pour qu'il affiche la représentation hexadécimale de 7531.

12, pour les plus rapides) **Écrire** un algorithme pyramide permettant d'imprimer le triangle suivant, le nombre de lignes étant donné par l'utilisateur.

```
           0
          012
         01234
        0123456
       012345678
      01234567890
     0123456789012
```