

C: TP 07 Programmation en C

DAKKAR Borhen-eddine

Lycée le Corbusier

BTS SN

October 30, 2020

1 Objectifs du TP

Dans ce TP nous allons découvrir la programmation modulaire. Nous traitons différentes situations où l'utilisation des **fonctions** est essentielle pour l'automatisation de nos programmes.

2 Logiciels à utiliser

Sous Linux les outils nécessaires à la compilation du C sont déjà présents, il s'agit de "gcc" et d'un éditeur de texte "gedit" par exemple.

3 Exercices

3.1 Exercice 1 :

1. Écrivez une fonction nommée **Val_abs()** qui accepte un nombre **double** en argument et affiche sa valeur absolue.
2. Testez votre fonction.

3.2 Exercice 2 :

1. Ecrivez une fonction nommée **mult()** qui accepte deux nombres flottant comme arguments, elle multiplie ces deux nombres et renvoie le résultat.
2. Testez votre fonction.

3.3 Exercice 3 :

1. Ecrivez une fonction qui produit un tableau des nombres de 1 à 10, leurs carrés et leurs cubes. Aucun argument ne doit être passé à la fonction et la fonction ne renvoie aucune valeur.
2. Testez votre fonction.

3.4 Exercice 4 :

1. Déterminez la sortie produite par le programme suivant:

```
#include <stdio.h>
main( )
{
    int cont;
    double temp;
    double temp_convert(double);//-- Entête de la fonction --//
    for (cont = 1; cont <= 4; ++cont)
    {
        printf("Entrer la temperature en Fahrenheit : ");
        scanf("%lf", &temp);
        printf("La temperature en Celsius est %.2f\n\n", temp_convert(temp))
    }
}
```

```

    }
    double temp_convert(double in_temp) //-- Définition de la fonction --//
    {
        return ( (5.0/9.0) * (in_temp- 32.0) );
    }

```

2. Réécrivez la fonction **temp_convert()** pour qu'elle accepte une température et un caractère comme arguments. Si le caractère passé à la fonction est la lettre **f**, la fonction doit convertir la température passée de **Fahrenheit** en **Celsius**, sinon la fonction doit convertir la température passée de **Celsius** à **Fahrenheit** ($F = (C \times 9/5) + 32$).

3.5 Exercice 5 :

1. Ecrivez un programme **main** qui déclare un tableau nommé **taux** : 6, 5, 7, 2, 7.5, 8.3, 8.6, 9.4, 9.6, 9.8, 10.0.
2. Ecrivez une fonction **show()** qui accepte le tableau **taux** comme argument et puis affiche les nombres de ce tableau.
3. Faites appel à la fonction **show()** dans le **main**.

3.6 Exercice 6 :

Ecrivez un programme C qui déclare trois tableaux unidimensionnels nommés **prix**, **quantite** et **montant**. Chaque tableau doit être déclaré dans le **main()** et doit être capable de tenir dix nombres doubles.

Les valeurs du tableau **prix** sont : 10.62, 14.89, 13.21, 16.55, 18.62, 9.47, 6.58, 18.32, 12.15, 3.98. Les valeurs du tableau **quantite** sont : 4, 8.5, 6, 7.35, 9, 15.3, 3, 5.4, 2.9, 4.8.

Votre programme doit passer ces trois tableaux à une fonction appelée **extend()**, qui devrait calculer les éléments du tableau **montant** comme le produit des éléments correspondants dans **prix** et **quantite** (par exemple, $montant[1] = prix[1] * quantite[1]$). Après **extend()** a mis les valeurs dans **montant**, les valeurs de ce tableau doivent être affichées depuis le **main()**.