

TD 3:

Préprocesseur – Programmation modulaire

Exercice 1:

Ecrire un programme pour définir, en macro, un nouveau type BOOL, qui ne serait qu'un entier, avec aussi deux macros: TRUE et FALSE qui valent 0 et 1. Puis l'utiliser pour déclarer une variable et lui affecter les valeurs TRUE ou FALSE pour conditionner sur son contenu afin d'afficher "vrai" ou "faux".

Exercice 2:

Ecrire un programme qui calcule le volume d'une sphère via une macro-fonction ayant comme paramètre le rayon (saisi par l'utilisateur).

- La macro-fonction appellera une autre macro calculant le cube.
- La valeur π à définir aussi par une macro constante
- Rappel: $V = (4\pi R^3)/3$

Exercice 3:

Ecrire un programme organisé en trois fichiers:

- arithmetique.c: contenant les opérations arithmétiques (addition, soustraction, multiplication et division) sous forme de fonctions.
- main.c: contenant la fonction main() où on demande la saisie de deux nombres puis le choix de l'opération à effectuer (selon un menu affiché). Puis le résultat sera affiché.
- arithmetique.h: fichier entête contenant la déclaration des fonctions à appeler.

Exercice 4:

Ecrire un programme organisé en quatre fichiers:

- Remplissage_matrice.c
 - par un nombre unique répété
 - par des nombres aléatoires
 - * pour générer un nombre aléatoire entre 0 et 100 on utilise la fonction de stdlib.h: `a=rand() % (101);`

- par saisie manuelle
- Affichage_matrice.c
 - Affichage en une seule ligne (lignes concaténées et séparées par “|”)
 - Affichage en une seule colonne (colonnes concaténées et séparées par “|”)
 - Affichage en matrice (lignes et colonnes)
- main.c
 - Demander à l'utilisateur le nombre de lignes et de colonnes de la matrice,
 - Lui demander de choisir le mode de remplissage (en menu) puis faire le traitement nécessaire selon chaque choix.
 - Lui demander de choisir le mode d'affichage (en menu) puis faire le traitement nécessaire selon chaque choix.
- Header.h: contenant les déclarations nécessaires