

TP 3

Adresse du cours : <https://axlbonnet.github.io/dut-gim-pres>

Consignes pour développer sur Dev-C++ :

- Sauvegardez tous vos programmes dans le lecteur réseau "W:"
- N'utilisez pas d'espace dans les noms de vos fichiers
- Changez les paramètres suivants dans "Outils>Options de l'éditeur>Général"
 - Activer "indentation automatique", "utiliser le caractère tab", "Retour arrière désindente" et "aide parenthèses/accolades relatives"
 - Désactiver "tabulations intelligentes"
- Utiliser des tabulations au début de toutes les lignes pour aligner ou décaler par rapport à la ligne précédente.

Pour tous les exercices suivants, écrivez d'abord l'algorithme sur papier, puis en C.

TP3-1

Obtenir un nombre entier et déterminer s'il vérifie les propriétés suivantes :

- il est plus grand que 100

- il est multiple de 17

Recommencez tant que ce n'est pas le cas

TP3-2

Calculer et afficher la surface d'un nombre inconnu de triangles

Les dimensions (longueur et hauteur) de chaque triangle doivent être saisies. Une saisie incorrecte (négative) d'une dimension arrêtera le programme.

TP3-3

Rechercher et afficher la plus grande valeur parmi n entiers saisis

L'utilisateur saisira d'abord le nombre "n" de valeurs qu'il veut saisir. On affichera le maximum à la fin de toutes les saisies.

TP3-4

Rechercher et afficher la plus grande valeur parmi un nombre indéterminé d'entiers saisis.

Un nombre négatif arrêtera la saisie.

On affichera le nombre de saisies effectuées (le dernier nombre négatif ne comptant pas). On affichera le maximum à la fin de toutes les saisies, mais seulement s'il y en a eu au moins une.

TP3-Bonus (en C directement si désiré)

Ecrire un jeu où l'on doit trouver un nombre inconnu (entre 1 et 1000) à partir des seules indications "trop grand" / "trop petit". D'abord le nombre pourra être saisi par votre voisin (essayez de trouver un moyen de le cacher). Vous pourrez ensuite le générer aléatoirement en :

- rajoutant "#include <time.h>" au tout début du programme
- exécutant une seule fois au début l'instruction "srand(time(NULL));"
- obtenant un nombre entier (de type int) aléatoire avec "nombre = rand();"
- manipulant ce nombre pour qu'il soit entre 1 et 1000