

Cahier de TP n°1

I Formats d'affichage et les instructions conditionnelles

1. Compilation, création d'un exécutable

Créer un répertoire TP1 et vous y placer. Ecrire, dans un fichier dont le nom se termine par .c, un programme C qui affiche votre nom et votre prénom

Compiler et exécuter ce programme.

Remarque : utiliser la fonction printf de la librairie standard.

2. Variables, type, lecture au clavier

Ecrire un programme utilisant deux variables entières, dont la valeur est fixée dans le programme, qui affiche la somme, la différence, le produit et le quotient des deux variables.

Remplacer la première variable entière par une variable réelle.

Transformer pour que la valeur des variables soit lue au clavier.

Que faut-il changer pour que la division ne soit effectuée que si la deuxième variable est non nulle.

Remarque : utiliser les fonctions scanf et printf de la librairie standard.

3. Multiples de 2 et de 3

Ecrire un programme C qui lit deux nombres entiers saisis au clavier, qui affiche ces deux nombres, leur somme, et indique si les nombres sont divisibles par 2 ou par 3.

On désire que sur une ligne, chaque nombre soit écrit sur au moins 3 caractères, suivi de 2 espaces, et, écrit en toutes lettres divisible par 2 ou divisible par 3 ou divisible par 2 et par 3.

4. Equation de droites dans le plan

- Ecrire l'algorithme permettant de déterminer les coefficients a et b de l'équation $y = ax + b$ d'une droite du plan à partir de 2 points n'ayant pas la même abscisse.
- Ecrire un programme qui affiche l'équation de la droite passant par deux points P et Q dont les coordonnées auront été saisies au clavier.
- Ecrire l'algorithme permettant de déterminer le point d'intersection de 2 droites.
- Transformer le programme précédent pour afficher le point d'intersection de 2 droites.

5. Mise sous forme de siècles, années, mois, jours

Ecrire un programme transformant un nombre de jours J entré par l'utilisateur en nombre de siècles, de mois, de semaines et de jours restant.

Pour simplifier le problème, on considérera que tous les mois ont 30 jours et toutes les années 360 jours.

L'affichage se fera sous la forme :

J jours <==> xx siecle xx annee xx mois xx semaine xx jours

On appliquera les règles d'accord du pluriel. ("5 annees" et non pas "5 annee")

6. Tarif jeune?

Ecrivez un programme qui demande l'âge de l'utilisateur et lui indique s'il a droit au tarif jeune (moins de 26 ans).

7. Tarif réduit?

Ecrivez un programme qui demande l'âge de l'utilisateur et lui indique s'il a droit au tarif réduit (moins de 26 ans ou plus de 60 ans).

8. Saison

Ecrivez un programme qui demande de taper p pour printemps, e pour été, a pour automne, h pour hiver et affiche le nom de la saison correspondante. Le programme retournera un message d'erreur si la lettre tapée ne correspond pas à une saison.

9. Madame, mademoiselle ou monsieur?

Ecrivez un programme qui affiche Bonjour madame, Bonjour mademoiselle ou Bonjour monsieur selon le sexe et l'état marital de la personne.

10. Rabais

Un client achète N pommes. Chaque pomme a le même prix P. S'il achète plus que 5 pommes il obtient un rabais de 5%. S'il achète entre 3 et 5 pommes, il obtient un rabais de 3%. S'il achète moins que 3 pommes, il n'y a pas de rabais.

Ecrivez un programme qui calcule le prix total.

11. Moyenne de notes

Ecrivez un programme qui lit trois notes d'examens et leurs coefficients, puis affiche la moyenne pondérée, en précisant Non reçu si la moyenne est inférieure à 10, admissible dans le cas contraire.

Ecrivez maintenant un programme qui lit quatres notes et affiche la moyenne des trois meilleures

II Instruction conditionnelle et boucle for

1. Trois entiers

Ecrire un programme qui calcule le minimum et le maximum de trois entiers choisis par l'utilisateur.

Ecrire un programme qui demande à l'utilisateur 3 entiers et qui les affiche en ordre croissant.

2. Extraction d'entiers

Ecrire un programme qui affiche tous les entiers impairs de l'intervalle [11, 119] qui ne sont divisibles ni par 3, ni par 5 ni par 7.

Votre programme devra aussi indiquer le nombre d'entiers affichés.

3. Suite d'entiers

Ecrire un programme qui lit un entier positif n ainsi que n valeurs entrées par l'utilisateur et calcule combien de ces valeurs sont positives, négatives ou nulles.

4. Puissance d'un entier

Ecrire un programme qui calcule a^n a est un entier quelconque et n est un entier positif ou nul. Les entiers a et n sont choisis par l'utilisateur et si n n'est pas positif ou nul un message d'erreur doit être envoyé.

5. Somme et somme alternée des puissances d'un entier

Ecrire un programme qui calcule $a^0 + a^1 + a^2 + \dots + a^n$, la somme des n premières puissances d'un entier a, où a et n sont des entiers comme dans l'exercice précédent.

Par exemple,

si $a = 2$ et $n = 5$, le programme doit afficher 63 puisque $2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 = 63$.

Modifier ce programme pour qu'il calcule la somme alternée des n premières puissances de a : $a^0 - a^1 + a^2 - \dots + a^n$.

Par exemple, si $a = 2$ et $n = 5$, le programme doit afficher -21

puisque $2^0 - 2^1 + 2^2 - 2^3 + 2^4 - 2^5 = 1 - 2 + 4 - 8 + 16 - 32 = -21$.

6. Affichage d'un rectangle

Ecrire un programme qui lit deux entiers positifs `long` et `large` entrés par l'utilisateur et affiche un rectangle plein de longueur `long` et de largeur `large`.

Vous modifierez ensuite le programme pour qu'il affiche un rectangle vide.

Par exemple, pour `long = 7` et `large = 5`, le premier rectangle est plein et le second est vide.

* * * * *	* * * * *
* * * * *	* * * * *
* * * * *	* * * * *
* * * * *	* * * * *
* * * * *	* * * * *