

Série de TP N°4 – Les instructions de test : Si...Fin-Si & Si...Sinon...Fin-Si

But de TP :

Le but du TP est de permettre aux étudiants de comprendre et de maîtriser les structures de contrôle conditionnelles en programmation.

Exercice N°01 : (Algorithme → Programme en langage C)

Soit l'algorithme suivant :

Algorithme exo1 ;

Variables

a, b, c, d: **entier** ;

Début

// Entrées

Écrire("Donner trois nombres entiers : ") ;

Lire(a,b,c) ;

// Traitements

Si (a<b) **alors**

Si (a<c) **alors**

 d ← a

Sinon

 d ← c ;

Fin-Si

Sinon

Si (b<c) **alors**

 d ← b

Sinon

 d ← c ;

Fin-Si

Fin-Si

// Sorties

Écrire ("Le résultat = ", d) ;

Fin.

Questions :

- 1- Traduire l'algorithme en un programme en langage C.
- 2- Compiler et exécuter le programme pour les valeurs suivantes :
 - a=1, b=2, c=4
 - a=2, b= -1, c=4
- 3- Déduire ce que fait cet algorithme ?
- 4- Dérouler l'algorithme/Programme pour les cas suivants :
 - a=1, b=2, c=4
 - a=2, b= -1 et c=4

Exercice N°02 :

Écrire un algorithme/programme en **langage C** qui permet de lire deux nombres entiers (**a, b**), calcule et affiche la valeur de **c** comme suit :

$$c = \begin{cases} 0 & \text{si } a = b \\ a.b & \text{si } a > b \\ a + b & \text{si } a < b \end{cases}$$

Exercice N°03 :

Écrire un programme en langage C qui permet de déterminer le prix d'un article en fonction de plusieurs critères :

1. Si le prix initial de l'article est inférieur à 100 D.A :
 - Si la quantité achetée est inférieure à 5, appliquer une remise de 5%.
 - Sinon, appliquer une remise de 10%.
2. Si le prix initial de l'article est compris entre 100 D.A et 500 D.A :
 - Si la quantité achetée est inférieure à 10, appliquer une remise de 15%.
 - Sinon, appliquer une remise de 20%.
3. Si le prix initial de l'article est supérieur à 500 D.A:
 - Appliquer une remise de 25% sur le prix initial.

Exercice N°04 :

Écrire un algorithme/programme en langage C qui permet d'afficher trois valeurs numériques A, B et C avec ordre croissant ?

Série de TP N°4 – Exercices supplémentaires

Exercice Sup-01 :

Écrire un programme en langage C qui permet à l'utilisateur de saisir un nombre entier. Le programme devra ensuite déterminer si ce nombre est pair ou impair.

Exercice Sup-02 :

Écrire un algorithme/programme Pascal qui permet d'introduire l'IMC (Indice de Masse Corporelle) d'une personne et d'afficher des informations concernant la catégorie de son IMC comme suit :

- ☞ « **Sous-poids** » Si $IMC < 18.5$
- ☞ « **Normal** » Si $18.5 \leq IMC \leq 27.0$
- ☞ « **Sur-poids** » Si $27.0 < IMC < 32.0$

Exercice Sup-03 :

Écrire un programme en langage C qui permet de résoudre l'équation du second degré $ax^2 + bx + c = 0$

Exercice Sup-04 :

On demande d'écrire l'algorithme d'une fiche de paie journalière d'un ouvrier rémunéré à la tâche. Pour cela, on donne :

- La valeur de cette rémunération par pièces réalisées VP,
- Le salaire brut (SB) est calculé selon le nombre de pièces correctes réalisées pendant la journée (NPC) comme suit :

Si $NPC \leq 100$, l'ouvrier touche $NPC * VP$

$NPC > 100$, l'ouvrier touche $150 * VP$

- On enlève à la fin 10% du salaire pour les charges sociales (CS).

Calculer et afficher le salaire journalier brut (SB), les charges sociales (CS) et salaire journalier net (SN). NB :

Salaire brut = salaire totale ; Salaire net = salaire sans les charges sociales.

Exercice Sup-05 :

Écrire un algorithme et sa traduction en programme C qui permet de calculer et afficher le nombre de : **Centaines, Dizaines et Unités** constituant un nombre entier « nb » ($0 < nb < 1000$)

Exemple :

- ☞ nb = 385, nb est constitué de 3 centaines, 8 dizaines et 5 unités
- ☞ nb = 93, nb est constitué de 9 dizaines et 3 unités
- ☞ nb = 4, nb est constitué de 4 unités

Exercice Sup-N°06 :

Écrire un algorithme/programme en langage C qui permet d'afficher trois valeurs numériques A, B et C avec ordre décroissant ?