

TP Structure des Ordinateurs et Applications

Série de TP N°4 – (Tests : SI...FIN-SI SI...SINON...FIN-SI)

Exercice N°01 : (Algorithmme → Programme Pascal) Soit l'algorithme suivant :

Algorithmme exo_01;

Variables

N : Entier;

Début

Ecrire ("Entrez Un Nombre Entier : ");

Lire (N);

Si (N = 0) Alors

Ecrire ("Le Nombre Est Nul Et Pair.");

Sinon

Si (N > 0) Alors

Si (N Mod 2 = 0) Alors

Ecrire ("Le Nombre Est Positif Et Pair.");

Sinon

Ecrire ("Le Nombre Est Positif Et Impair.");

Finsi;

Sinon

Si (N Mod 2 = 0) Alors

Ecrire ("Le Nombre Est Négatif Et Pair.");

Sinon

Ecrire ("Le Nombre Est Négatif Et Impair.");

Finsi;

Finsi;

Finsi;

Fin.

Questions:

1. Traduisez l'algorithme ci-contre en un programme Pascal.
2. Compiler et exécuter le programme pour:
 - N= 12;
 - N= - 9;
 - N= 0.
3. Dérouler l'algorithme pour: N= - 9.
4. Proposez un algorithme (Organigramme) de cet algorithme.

Exercice N°02 :

- 1- Écrire un programme en **Pascal** qui demande à l'utilisateur de saisir **trois nombres entiers**. Le programme doit afficher **le plus grand** des trois.
- 2- Donnez son Organigramme (Algorigramme)

Exercice N°03 : Une boutique souhaite gérer le stock d'un produit. Ecrire un programme en pascal qui permet de :

- Lire la quantité initiale en stock, la quantité de produit entrant (réapprovisionnement) et la quantité de produit sortant (ventes ou sorties).
- Calculer le stock restant :

Le Stock restant = Stock initial + Entrée - Sortie

Ensuite Vérifie et afficher un message selon les conditions suivantes :

- ☞ Si le stock restant < 0 → "Erreur: stock insuffisant !"
- ☞ Si le stock restant = 0 → "Stock épuisé !"
- ☞ Si le stock restant ≤ 10 → "Stock faible : reste X produits"
- ☞ Sinon → "Stock suffisant : reste X produits"

Exercice N°04 :

On souhaite calculer la **facture d'un client** selon le nombre de produits achetés et appliquer une remise selon le total, écrire un programme en pascal qui permet de :

1. Lire le **prix unitaire** du produit et la **quantité achetée**.
 2. Calculer le **montant total** : Total = Prix_unitaire × Quantité
- Appliquer une **remise** selon les conditions suivantes :

- Si Total < 100 DA → pas de remise
- Si $100 \leq \text{Total} \leq 500$ DA → remise de 5%
- Si Total > 500 DA → remise de 10%

3. Afficher le **montant total**, la **remise appliquée** et le **total à payer**.

TP Structure des Ordinateurs et Applications

Exercices supplémentaires – SÉRIE DE TP N°04

Exercice sup-01 : (Algorithme → Programme Pascal)

Écrire un algorithme et sa traduction en programme Pascal qui permet de calculer et afficher le nombre de **centaines**, **dizaines** et **unités** constituant un nombre entier « **nb** » ($0 < \text{nb} < 1000$).

Exemple : $\text{nb} = 385$, nb est constitué de 3 centaines, 8 dizaines et 5 unités

$\text{nb} = 93$, nb est constitué de 9 dizaines et 3 unités

$\text{nb} = 4$, nb est constitué de 4 unités

Exercice sup-02 : (Fonctions Trigonométriques)

1/ Écrivez un programme en langage Pascal nommé (**TrigonometricFunction**) qui demande à l'utilisateur d'entrer un angle en degrés, le code convertit en radians, puis calcule et affiche la valeur de la fonction sinus si l'angle est compris entre 0 et 180 degrés, la somme de la fonction sinus et de la fonction cosinus si l'angle est supérieur à 180 degrés, ou la fonction cosinus si l'angle est négatif. Voir la fonction suivante :

$$y = \begin{cases} \cos(x) & \text{si } x < 0 \\ \sin(x) & \text{si } 0 \leq x < \pi \\ \cos(x) + \sin(x) & \text{si } x \geq \pi \end{cases}$$

2/ Donner son algorithme (Organigramme).

Exercice sup-03 :

Soient X, Y et Z trois nombres entiers, écrire un algorithme et son algorithme ainsi sa traduction en programme Pascal, intitulé **ordre_croissant**, qui permet d'afficher les trois valeurs X, Y et Z dans l'ordre croissant.

Exercice sup-04 :

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 $\text{IMC} < 18.5$

☞ « **Normal** » Si $18.5 \leq \text{IMC} \leq 27.0$

☞ « **Sur-poids** » Si $27.0 < \text{IMC} < 32.0$

Exercice sup-05 :

Pour simuler les ventes, un magasin offre un taux de réduction sur achat dans les conditions suivantes

- Si le montant d'achat $m_achat > 3000$ DA, le taux de réduction est de 30%.
- Si $2000 \leq m_achat \leq 3000$ DA, le taux de réduction est de 20%.
- Si $1000 \leq m_achat < 2000$ DA, le taux de réduction est de 10%.
- Si $m_achat < 1000$ DA, le taux de réduction est de 5%.

Écrire un algorithme et un programme Pascal qui calcule et affiche le prix total à payer, sachant que le prix total à payer (prix_total) est calculé à partir du montant d'achat introduit et le taux de réduction défini dans chaque condition.

Exercice sup-06:

Écrire un algorithme et un programme Pascal qui calcule, affiche la valeur de **X** comme suit :

$$X = \frac{1}{2} \times A \quad \text{Avec} \quad A = \begin{cases} \frac{2 + N^2 + 3N}{2} & \text{si } N \geq 0 \\ \frac{N^2 + |N|}{2} & \text{sinon} \end{cases}, N \text{ est nombre entier}$$