

TD 6 : LES ENREGISTREMENTS

OBJECTIFS :

Maitriser les enregistrements.

Exercice 1

Un étudiant est défini par les caractéristiques suivantes : numIns, nom, prenom, dateNaiss (définie par le jour, le mois et l'année de naissance), tel, moyenne.

- 1- Donner la déclaration d'un enregistrement Etudiant,
- 2- Ecrire la fonction saisieEtudiant() : Etudiant
et la procédure afficheEtudiant(E : Etudiant)
- 3- Définir le type Tab un tableau de 30 étudiants.
- 4- Ecrire une procédure SaisieGroupe (T : Tab , n :entier) qui saisie un tableau de n étudiants.
- 5- Ecrire une procédure AfficheGroupe (T : Tab , n :entier) qui affiche un tableau de n étudiants.
- 6- Ecrire une fonction meilleurEtudiant (T : Tab , n :entier) qui affiche l'étudiant ayant la moyenne la plus élevée.
- 7- Ecrire une fonction qui calcule la moyenne de toute la classe.
- 8- Ecrire une procédure qui affiche les étudiants nés en l'année 2000.

Exercice 2

Un nombre complexe Z est entièrement défini par ses parties réelle a et imaginaire b ($Z = a + bi$).

- 1- Donner la déclaration d'un nombre complexe,
 - 2- Ecrire les fonctions : ReelZ, ImagZ et Module donnant les attributs d'un nombre complexe respectivement : la partie réelle, la partie imaginaire et le module),
 - 3- Ecrire les actions paramétrées : SommeZ, DiffZ et ProdZ nécessaires à l'arithmétique sur les complexes, respectivement pour l'addition, la soustraction et la multiplication,
 - 4- Ecrire une procédure ConjZ qui calcule le conjugué d'un nombre complexe.
 - 5- Ecrire une fonction EgaleZ qui teste l'égalité de deux nombres complexes.
 - 6- Ecrire une procédure EcrireZ qui permet d'afficher un nombre complexe.
- Soit TC un tableau de N nombres complexes ($N \leq 100$). En utilisant les actions paramétrés précédentes, écrire un algorithme qui :
- Affiche l'élément de TC ayant le plus grand module. Puis vérifie l'existence de son conjugué dans TC.
 - Calcule la somme Zs et le produit Zp des éléments non nuls du tableau TC.
 - Calcule et affiche la différence entre Zs et Zp si elle est imaginaire pur.

Exercice 3

Soit Tdate un type date composé des champs entiers JJ,MM,AA.

- Ecrire une AP CompareD permettant de comparer deux dates D1 et D2.
- Soit TD un tableau de N dates ($N \leq 100$). En utilisant l'AP CompareD, écrire un algorithme permettant de trier ce tableau dans l'ordre croissant des dates.

Exercice 4

Ecrire une fonction qui détermine la différence en nombre de jours entre deux dates.

Exercice 5

Considérons les types d'enregistrements suivants :

Type

TDate = Enregistrement

Jour, mois, année : entier ;

FinEnregistrement;

TAdresse = Enregistrement

Numéro : entier ;

Rue : chaîne [50] ;

Ville : chaîne [20] ;

gouvernorat : chaîne [20] ;

Cw : entier ;

{ Code gouvernorat }

FinEnregistrement;

THabitant = Enregistrement

Nom, prenom : chaîne [20] ;

Date_naiss : date ;

Residence : Adresse ;

FinEnregistrement;

Ecrire un algorithme permettant de :

- 1- Remplir un tableau T de N habitants ($N \leq 100$).
- 2- Afficher à partir de T les adresses des habitants nés avant une année de naissance donnée.
- 3- Afficher les noms et les dates de naissance des habitants de la ville de 'eljem' du gouvernorat de 'mahdia'.
- 4- Editer le nombre d'habitants par gouvernorat.

Exercice 6

On s'intéresse à la gestion des véhicules d'un parc auto. Chaque véhicule est caractérisé par une matricule, une marque, un modèle, une couleur, le nombre de places, une puissance fiscale.

- 1- Donner l'enregistrement permettant de décrire un véhicule.
- 2- Décomposer le matricule en ses composants élémentaires puis donner la nouvelle structure de l'enregistrement.
- 3- Ecrire un algorithme qui permet de :
 - Stocker les informations d'un parc auto regroupent au max 50 véhicules en utilisant les structures adéquates ;
 - Etablir la liste (matricule, marque, modèle, puissance) des véhicules d'une couleur donnée ;
 - Etablir un tableau statistique contenant le nombre de véhicules par marque;