



Rappel :

- Toute donnée en mémoire externe est organisée sous forme de Fichier(s).
- Un Fichier est :
 - ✓ une séquence d'octets,
 - ✓ repéré par un nom (dit nom externe), par exemple "texte.txt" ou "TP05-S2.c",...
 - ✓ enregistré sur un support physique non volatile de l'ordinateur : disque, clé USB, carte sd,...
- Un Fichier n'est pas détruit à l'arrêt de l'ordinateur.
- La taille d'un Fichier n'est pas précisée à sa création.

Exercice n° :01

Considérons le type enregistrement suivant :

Type Etudiant = **Enregistrement**

Matricule : entier ;

Nom, Prenom : chaîne [20] ;

Moyenne : réel ;

FinEnregistrement;

Soit **T** un tableau de **N** étudiants ($N \leq 100$).

- ✓ Ecrire un algorithme permettant de recopier tous les étudiants admis appartenant à **T** dans un fichier **ADMIS** de type étudiant. Un étudiant est admis si sa moyenne est supérieure ou égale 10.

Exercice n° :02

Soient les enregistrements suivants :

Type Date = **Enregistrement**

Jour, mois, année : **entier** ;

FinEnregistrement;

Type Etudiant = **Enregistrement**

Nom, Prenom : chaîne [20] ;

Date_Naiss : Date ;

Filiere : TDiscipline ;

FinEnregistrement;

Type TDiscipline = **Enregistrement**

Discipline : chaîne [10] ;

Faculté : chaîne [20] ;

FinEnregistrement;

Soit **FEtudiant** un fichier d'étudiants. Ecrire un algorithme qui permet de :

- Remplir le fichier **FEtudiant**.
- Eclater le fichier **FEtudiant** en deux fichiers, **F1** (étudiants de la faculté "FSC") et **F2** (étudiants des autres facultés).

Exercice n° :03

Soit le type suivant :

<p>Type Produit = Enregistrement Code : Entier ; Désignation : Chaîne [100] ; Prix : Réel ; FinEnregistrement;</p>

Soit **F** un fichier de produits.

- Ecrire une **Fonction** qui vérifie si les éléments de **F** sont triés par ordre croissant de leur Code.

Exercice n° :04

1- Soient **F1** et **F2** deux fichiers d'entiers strictement positifs et sans répétition.

- Ecrire un algorithme qui construit (créé) un fichier (**G**) d'entiers tel que **G** contient pour chaque valeur de **F1** la valeur et tous ses multiples appartenant à **F2** (**F1** et **F2** sont supposés existants).

Exemple :

F1 : 3 10 20 17

F2 : 3 6 19 60 40 30

G : 3 3 6 60 30 10 60 40 30 20 60 40 17

2- Ecrire un algorithme qui permet à partir du fichier résultat (**G**) de générer un autre fichier (**H**) contenant toutes les valeurs du fichier (**G**) (sans répétition) avec leur nombre.

Exemple :

H : 3 2 6 1 60 3 30 2 10 1 40 2 20 1 17 1

Exercice n° :05

Soient **F1** et **F2** deux fichiers de chaînes de caractères. Chaque chaîne représente un mot.

- Ecrire un algorithme qui construit (créé) un fichier **F3**, tel que **F3** contient les mots de **F1** qui n'existent pas dans **F2**.



Exercice n° :01

Algorithme Etudiants_ADMIS ;

Type Etudiant = **Enregistrement**

Matricule : **Entier** ;

Nom, Prenom : **chaîne** [20] ;

Moyenne : **Réel** ;

FinEnregistrement;

Var T : Tableau[1..100] de type **Etudiant** ;

F : **Fichier** de type **Etudiant** ;

I, N : **Entier** ;

Debut

/ lecture des éléments du tableau de type Etudiant */*

Repeter

Ecrire("Donner le nombre d'étudiants") ;

Lire(N) ;

Jusqu'à (N>0 et N≤100) ;

Pour I de 1 à N **Faire**

Lire(T[I].Matricule) ;

Lire(T[I].Nom , T[I].Prenom) ;

Lire(T[I].Moyenne) ;

FinPour ;

*/*création du fichier des admis */*

Assigner(F,"ADMIS") ;

Réécrire(F) ;

/ Assigner et Ouvrir le fichier F en écriture */*

Pour I de 1 à N **Faire**

Si (T[I].Moyenne)≥10 **Alors**

Ecrire(F,T[I]) ;

/ Ecrire les Etudiants Admis dans le Fichier F */*

FinSi;

FinPour ;

Fermer(F) ;

Fin.

Exercice n° :02

Algorithme Eclate_Fichier ;

Type Date = Enregistrement
Jour, mois, annee : entier ;

FinEnregistrement;

Type TDiscipline = Enregistrement

Discipline : chaîne [10] ;

Faculte : chaîne [20] ;

FinEnregistrement;

Type Etudiant = Enregistrement

Nom, Prenom : chaîne [20] ;

Date_Naiss : Date ;

Filiere : TDiscipline ;

FinEnregistrement;

Var

F, F1, F2 : Fichier de type Etudiant ;

FSC, AUTRE : Booléen ;

Debut

Assigner(F,"Fetudiant") ;

Réécrire(F) ; */* Assigner et Ouvrir le fichier F en écriture */*

*/*Lire 1er Nom pour le premier Etudiant à l'extérieur de la boucle */*

Ecrire("Donner le Nom de l'étudiant :") ; Lire(Etudiant.Nom) ;

Tantque (Etudiant.Nom ≠ "") Faire

Ecrire("Donner le Prénom de l'étudiant :") ;

Lire(Etudiant.Prenom) ;

Ecrire("Donner la date de naissance de l'étudiant :") ;

Lire(Etudiant.Date_Naiss.Jour , Etudiant.Date_Naiss.mois , Etudiant.Date_Naiss.Année) ;

Ecrire("Donner la Discipline et la Faculté :") ;

Lire(Etudiant.Filiere.Discipline , Etudiant.Filiere.Faculté) ;

Ecrire(F,Etudiant) ; */* Ecrire l'étudiant dans le Fichier F */*

Ecrire("Donner le Nom de l'étudiant suivant :") ;

Lire(Etudiant.Nom) ; */*Lire le nom de l'étudiant suivant */*

FinTantque ;

Fermer(F) ;

/ Fermer le fichier F */*

Relire(F) ;

/ Ouvrir le Fichier F en Lecture */*

Assigner(F1," F_FSC ") ;

Réécrire(F1) ; */* Assigner et Ouvrir le Fichier F1 en Lecture */*

Assigner(F2, "F_AUTRE");

Réécrire(F2); */* Assigner et Ouvrir le Fichier F2 en Lecture */*

Si (FDF(F)) **Alors**

Ecrire("Fichier F est vide");

Sinon

Tantque (Non FDF(F)) **Faire**

Lire(F,Etudiant) ;

Si (Etudiant.Filiere.Faculté = " FSC ") **Alors**

Si (Non (FSC)) **Alors**

Assigner(F1,"F_FSC") ;

Récrire(F1) ;

FSC ←Vrai ;

FinSi ;

Ecrire(F1,Etudiant) ;

Sinon

Si (Non (Autre)) **Alors**

Assigner(F2,"F_AUTRE");

Récrire(F2);

AUTRE←Vrai ;

FinSi ;

Ecrire(F2,Etudiant) ;

FinSi;

FinTantque ;

Fermer(F) ;

Si (FSC) **Alors**

Fermer(F1) ;

FinSi;

Si (AUTRE) **Alors**

Fermer(F2) ;

FinSi;

FinSi ;

Fin.

Exercice n° :03

Fonction TrierFichier(F :fichier de type Produit) :**booléen** ;

Var

Eprod :**Produit** ;

Code :**Entier** ;

Ftrie : **booléen** ;

Debut

Assigner(F,"Produit.txt ") ;

Relire(F) ;

/ Assigner et ouvrir le Fichier F en Lecture */*

Ftrie ←Vrai ;

2-

Algorithme Valeur_Répétitive ;

Var

G, H, G1, G2 : Fichier de entier ;

X, Y , Nbr-Rep :entier :

Debut

Assigner(G,"File3") ;

Assigner(H, "FileH") ;

Assigner(G1, "Inter1") ;

Assigner(G2, "Inter2") ;

Relire(G) ;

/ ouvrir le Fichier G en Lecture */*

Recrire(G1) ;

/ ouvrir le Fichier G1 en Ecriture */*

*/*Copie le fichier G dans le fichier G1, on utilise des fichiers intermédiaires */*

Tantque (Non FDF(G)) **Faire**

Lire(G,X) ;

Ecrire(G1,X) ;

FinTantque ;

Fermer(G) ;

Fermer(G1) ;

Relire(G1) ;

/ ouvrir le fichier G1 en lecture */*

Recrire(H) ;

/ ouvrir le fichier H en Ecriture */*

Tantque (Non FDF(G1)) **Faire**

Lire(G1,X) ;

Ecrire(H,X) ;

/ Traiter les répétitions et créer G2 (non traitées)*

Recrire(G2) ;

/ ouvrir le fichier G2 en Ecriture */*

Nbr-Rep ← 1 ;

Tantque (Non FDF(G1)) **Faire**

Lire(G1,Y) ;

Si (X=Y) **Alors**

Nbr-Rep ← Nbr-Rep + 1 ;

Sinon

Ecrire(G2,Y) ;

FinSi ;

FinTantque ;

*/*Ecrire la répétition de X dans le fichier H */*

Ecrire(H, Nbr-Rep) ;

Fermer(G1) ;

Fermer(G2) ;

*/*Copie de G2 dans G1 Pour traiter le reste. Ecraser l'ancien G1 */*

Relire(G2) ;

Recrire(G1) ;

Tantque (Non FDF(G2)) **Faire**

Lire(G2,X) ;

Ecrire(G1,X) ;

FinTantque ;

Fermer(G1) ;

Fermer(G2) ;

*/*Ré ouvrir G1 en lecture (non traitées) */*

Relire(G1) ;

FinTantque ;

Fermer(G1) ;

Fermer(H) ;

Fin.

Exercice n° :05

Algorithme MotF1F2F3 ;

Var

F1, F2 , F3 : fichier de chaine[50] ;

X ,Y :Chaine[50] ;

Trouve :**booleen** ;

Debut

Assigner(F1, "File1") ;

Assigner(F2, "File2") ;

Assigner(F3, "File3") ;

Relire(F1) ;

/ ouvrir le fichier F1 en Lecture */*

Reecrire(F3) ;

/ ouvrir le fichier F3 en Ecriture */*

Tantque (Non FDF(F1)) **Faire**

Lire(F1,X) ; */*Lire un mot de fichier F1 */*

Trouve←Faux ; */*on suppose que le mot n'existe pas dans F2 */*

Relire(F2) ; */*revenir à chaque itération au début du fichier F2 */*

Tantque (Non FDF(F2) **et** Non (Trouve))**Faire**

Lire(F2, Y) ;

Si (Y=X) **Alors**

Trouve ←Vrai ;

FinSi ;

FinTantque ;

/ Si le mot n'est pas trouvé après le FDF de F2 on le met dans F3 */*

Si (Non (Trouve)) **Alors**

Ecrire(F3,X) */*Ecrire un mot dans le fichier F3 */*

FinSi;

Fermer(F2) ;

FinTantque ;

Fermer(F1) ;

Fermer(F3) ;

Fin.