

Série d'exercices : Les fichiers (files)

1. Soit le fichier **NOMBRES.BIN** qui contient une liste de nombres entiers. Écrire un algorithme qui affiche les nombres du fichier, leur somme et leur moyenne.
2. Écrire un algorithme qui crée le fichier **MOTS.TXT** contenant une série de mots (longueur maximale d'un mot: 20 caractères). La saisie des mots se terminera à l'introduction du symbole '*' qui ne sera pas écrit dans le fichier.
 - Écrire un algorithme qui affiche le nombre de mots ainsi que la longueur moyenne des mots contenus dans le fichier **MOTS.TXT**.
 - Écrire un algorithme qui crée un deuxième fichier **MOTS10.TXT** contenant les mots du fichier **MOTS.TXT** de plus de 10 caractères.
3. Considérons le type enregistrement suivant :

```
Type Etudiant = Enregistrement
    Matricule : entier ;
    Nom, Prenom : chaine [20] ;
    Moyenne : réel ;
Fin;
```

Soit **T** un tableau d'au plus 100 étudiants.

- 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.
4. Soient les enregistrements suivants :

```
Type TDate = Enregistrement Jour, mois, année : entier ; Fin;
TDiscipline = Enregistrement
    Discipline : chaine [10] ; Faculté : chaine [20] ; Fin;
TEtudiant = Enregistrement
    Nom, prenom : chaine [20] ;
    DateN : TDate ;
    Filiere : TDiscipline ;
Fin;
```

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 'FEI') et **F2** (étudiants des autres facultés).

5. Soient **F1** et **F2** deux fichiers d'entiers strictement positifs et sans répétition. Ecrire un algorithme qui construit (crée) un fichier **G** d'entiers tel que **G** contient pour chaque valeur de **F1** la valeur et

Série d'exercices : Les fichiers (files)

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
```

- 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
```

6. Soit **F** un fichier d'entiers représentant des séquences de nombres séparées par un ou plusieurs zéro. Ecrire un algorithme qui réalise les traitements suivants :
- A partir de **F** (fichier existant), crée un fichier **G** contenant pour chaque séquence, la moyenne des nombres qui la constituent.
 - Puis, Supprimer les valeurs nulles du fichier **G**.

Exemple :

```
F : 0 0 1 4 3 7 0 0 0 6 -9 2 7 -6 0 -10 3 0 0
G : 3,75 0,00 -3,50 Avant suppression
G : 3,75 -3,50 Après suppression
```

7. Soit le type suivant :

```
Type Produit = Enregistrement
Code : Entier ;
Désignation : Chaîne [ 80 ] ;
Prix : Réel ;
Fin ;
```

Soit **F** un fichier de produits. Ecrire une **Fonction** qui vérifie si les éléments de **F** sont triés par ordre croissant de leur Code.

8. Soit **Fmot** un fichier de caractères alphabétiques contenant des mots séparés par un ou plusieurs caractères blanc.
- Ecrire une fonction *Palindrome* qui vérifie si un mot donné est un mot palindrome.
 - Ecrire un algorithme qui affiche le nombre de mots palindromes et le plus court mot palindrome.