Les Fichiers :

Introduction :

Un fichier est une structure informatique qui permet le stockage permanent des données sur un support externe.

Les données sont stockées selon deux formats possibles :

* Texte : les données sont écrites sous format texte, une suite de caractères. Ces fichiers sont éditables, et donc modifiables avec des éditeurs de texte. Le nombre réel 12.5 est écrit aves les caractères ‘1’, ‘2’,’.’,’5’.
* Binaire : les données sont écrites selon leur représentation en mémoire centrale. Le nombre 12.5 est écrit selon sa représentation en mémoire centrale sur un format de 4 octets soit 8 bits (0/1). Ces fichiers ne sont pas éditables. Il faut utiliser des logiciels spécifiques pour les visualiser.

L’accès aux données se fait selon deux types :

* De façon séquentielle : Avant d’exécuter une donnée cible, il faut lire toutes les données qui se trouvent avant celle-ci.
* De façon direct. Pour accéder à la donnée cible, on réalise un déplacement ou un saut de toutes les données se trouvant avant elle. Donc on doit calculer la taille du saut en octets à réaliser.

I – Opérations sur les fichiers

1. La déclaration :

Algo : nomfichier : fichier

Langage C : File \* nomfichier : // on déclare un pointeur de fichier, adresse mémoire

1. L’ouverture

Avant toute utilisation un fichier doit être ouvert en faisant la correspondance entre fichier logique déclaré par le programmeur et fichier logique qui existe dqns le systéme d’exploiration.

Algo : monfichier 🡨 ouvrir (nomPhysique, mode)

Exemple de nom physique : moncours.txt

Mode : lecture / ecriture / lecture

Langage C : nomfichier = fopen (nomPhysique, mode)

Mode : ‘r’ , ‘w’, ‘a’ read, write, append

1. La fermeture :

Tout fichier ouvert doit être fermé après utilisation

Algo : fermer (nomfichier )

Langage C : fclose(nomfichier) ;

1. Ecriture et lecture

Algo : ecrire  (nomfichier, nomvariable)

Lire (nomfichier, nomvariable)

Langage C :

Lecture / Ecriture d’un caractère == fgetc, fpuc

Lecture / Ecriture d’une chaine == fgets / fputs

Lecture / Ecriture formaté (%d, %f, %s etc) fprintf, fscanf

Lecture / Ecriture binaire : fwrite, fread

1. Fin de fichier

La fin de fichier est matérialisée par le caractère OEF.

Algo : fin\_fichier (nomfichier) retourne vrai / faux

Langage C : foef (nomfichier) ; retourne aussi un bool