Chapitre 1 :
Les fichiers

Introduction

- Lors de l'exécution d'un programme, toutes les données introduites par l'utilisateur seront stockées dans la mémoire principale de l'ordinateur (RAM).
- A la fermeture du programme elles seront perdues.
- Il est parfois nécessaire de conserver ces données, d'où on a réfléchi à les stocker d'une manière permanente sur un support de stockage physique en utilisant les fichiers.

Définition d'un fichier

- Un fichier est un ensemble structuré de données de même type, nommé et enregistré sur un support de stockage permanent (disque dur, flash disque, CD Rom..)
- Le fichier possède **un nom physique** connu par le système d'exploitation (avec lequel il est enregistré sur le support) et **un nom logique** (le nom de la variable utilisé au niveau du programme).

Les types d'accès aux fichiers

Il existe deux types d'accès aux données d'un fichier :

- **Accès séquentiel :** Pour lire une information, il faut passer par toutes les informations qui la précédent.
- **Accès direct :** On peut accéder directement à l'information en précisant son numéro d'ordre.

NB : C'est au niveau du programme qu'on choisit le type d'accès.

Les types des fichiers

On trouve deux types de fichiers : les fichiers de données et les fichiers textes.

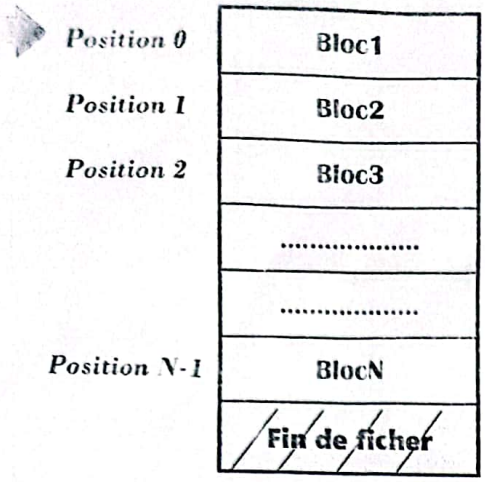
1. Les fichiers de données (ou fichiers binaires ou fichiers typés) :

a. Définition

Un fichier binaire contient des éléments de même type (fichier d'entiers, fichier de réels, fichier de caractères, fichier de chaînes, fichier d'enregistrements).

Dans un fichier binaire, les données sont présentées comme elles sont stockées dans la mémoire vive : une suite d'octets.

- Un fichier de données n'est pas lisible directement : son contenu n'est lisible que par le programme qui l'a conçu.
- Les fichiers de données représentent un format de stockage des données compact et rapide en lecture et écriture.
- Dans un fichier de données, l'accès aux données peut être séquentiel ou direct.
- Un fichier de données est organisé en blocs.
- Les blocs de données formant le fichier de données sont enregistrés les uns à la suite des autres de façon linéaire.
- La position du premier bloc est toujours 0.
- Pour accéder aux blocs d'un fichier de données, on utilise un pointeur.



b. Déclaration d'un fichier de données

Syntaxe :

TDNT :

Type
NomType = Fichier de Type-element

TDO :

Objet	Type/Nature
NomVariable	NomType

Exemple :

TDNT :

Type
Fich = Fichier de réels

TDO :

Objet	Type/Nature
F	Fich

c. Les fonctions et les procédures sur les fichiers de données

Pour traiter les fichiers typés, on a recourt aux fonctions et procédures suivantes :

En algorithmique	En Python	Rôle
Ouvrir ("Chemin/NomPhysique" , NomLogique, "Mode")	NomLogique = open ("Chemin/NomPhysique" , "Mode")	Ouverture d'un fichier. Mode d'ouverture d'un fichier : - **"rb"** : Lecture - **"wb"** : Ecriture (création) - **"ab"** : Ecriture à la fin du fichier
Lire (NomLogique , Objet)	Objet = **pickle.load** (NomLogique)	Lecture d'un objet à partir d'un fichier.
Ecrire (NomLogique , Objet)	**pickle.dump** (Objet, NomLogique)	Ecriture d'un objet dans un fichier.
Fin_Fichier (NomLogique)	**Pas de correspondance.**	Test de fin fichier. Retourne **VRAI** si le pointeur est à la fin du fichier sinon elle retourne **FAUX**.
Fermer (NomLogique)	NomLogique.close()	Fermeture du fichier.

⇨ A chaque action **de lecture ou écriture** le pointeur **se déplace automatiquement** vers le bloc suivant.

⇨ En Python, pour utiliser les fonctions citées ci-dessus, il faut importer la bibliothèque **pickle**.