

L'utilisation d'un fichier se fait généralement en trois temps :

1. Ouverture du fichier
2. Utilisation (lecture/écriture)
3. Fermeture du fichier

```
#include <stdio.h>
FILE * f;
f = fopen("toto.txt", "rw");
```

- Renvoie un pointeur vers un descripteur de fichier (type FILE*).
- En cas d'échec, `fopen` renvoie NULL

Les principaux modes d'ouverture sont :

- `r` : lecture
- `w` : écriture
- `a` : écriture en fin de fichier (append)
- `b` : mode binaire (à ajouter au mode d'ouverture)

1 Fichiers textes

Les deux fonctions `printf` et `scanf` possèdent leur pendant pour les fichiers : `fprintf` et `fscanf`. Ces deux dernières fonctions prennent en premier argument supplémentaire un pointeur vers le descripteur de fichier.

L'exemple qui suit ouvre le fichier `data.txt`, calcule la somme des nombres à virgule qui s'y trouvent, puis ajoute ce total à la fin du fichier `somme.txt`.

```
#include <stdio.h>
int main(void) {
    FILE * in, * out;
    float sum=0, val=0;
    in=fopen("data.txt", "r");
    if (in==NULL) {
        printf("L'ouverture du fichier d'entrée a échoué\n");
        return -1;
    }
    while(fscanf(in, "%f", &val) != EOF) {
        sum+=val;
    }
    fclose(in)
    out=fopen("somme.txt", "a");
    if (out==NULL) {
        printf("L'ouverture du fichier de sortie a échoué\n");
        return -1;
    }
    fprintf(out, "%f", sum);
    fclose(out);
}
```

Analysez attentivement le lignes :

```
while(fscanf(in, "%f", &val) != EOF) {
    sum+=val;
}
```

`fscanf(in, "%f", &val)` est une instruction qui lit un nombre flottant à l'emplacement courant du fichier et le place dans `val`. Mais c'est aussi une instruction qui renvoie le nombre de valeurs lues (ici 1) ou bien le code EOF si le fichier est terminé.

La seule ligne `while(fscanf(in, "%f", &val) != EOF)` permet donc de lire chaque valeur flottante dans le fichier, stockant temporairement chaque valeur dans `val`, jusqu'à la fin du fichier.

2 Fichiers binaires

Les deux fonctions `fread` et `fwrite` permettent de lire et écrire des octets dans un fichier. La fonction `fread` prend en premier paramètre un pointeur vers la zone mémoire où inscrire les données lues, en second paramètre le nombre de données à lire, en troisième paramètre la taille d'une donnée en octets, et en dernier paramètre, un pointeur vers le descripteur de fichiers.

La fonction `fwrite` prend en premier paramètre un pointeur vers la zone mémoire où prendre les données à écrire dans le fichier, en second paramètre le nombre de données à écrire, en troisième paramètre la taille d'une donnée en octets, et en dernier paramètre, un pointeur vers le descripteur de fichiers.

Pour pouvoir faire des lectures/écritures binaires, il faut penser à suffixer le mode d'ouverture du fichier (`fopen`) par `b`.