

Langage C

Les fichiers

L2 Mathématique et Informatique

Université de Marne-la-Vallée

Manipulation de fichiers

Les fonctions d'entrées-sorties de la bibliothèque standard manipulent les fichiers via une structure de type `FILE`.

La position de l'index de lecture ou d'écriture dans le fichier est une des informations mémorisées dans cette structure. Cette position étant appelée à changer, les fonctions utilisent des pointeurs `FILE *`.

Trois `FILE *` sont ouverts par le shell:

- ▶ `stdin` entrée standard, ouvert en lecture,
- ▶ `stdout` sortie standard, ouvert en écriture,
- ▶ `stderr` sortie erreur standard, ouvert en écriture, non bufferisé.

Pour les autres fichiers, il faut d'abord obtenir un FILE * à l'aide de la fonction

FILE * fopen(const char * reference, const char *mode)

- ▶ reference est le chemin d'accès au fichier. Si le fichier n'existe pas alors il est créé *si et seulement si* le mode est un mode d'écriture et l'utilisateur a l'autorisation d'écriture dans le répertoire.
- ▶ mode indique le mode d'ouverture.
 - ▶ "r" ouverture en lecture, l'index de lecture est placé au début du fichier
 - ▶ "w" ouverture en écriture au début du fichier, le contenu précédent du fichier est perdu.
 - ▶ "a" ouverture en écriture à la fin du fichier,
 - ▶ "r+", "w+", "a+" ouverture en écriture et lecture respectivement au début, au début avec écrasement et à la fin, du fichier.

La fonction renvoie un pointeur sur la structure FILE associée au fichier ouvert ou NULL en cas d'échec.

Le nombre de pointeurs sur fichiers utilisables par un processus est limité.

Il est donc conseillé de fermer les fichiers sur lesquels le traitement est terminé.

La fonction `int fclose (FILE *stream);` effectue cette fermeture.

LECTURE ET ECRITURE DANS UN FICHER

Entrée sortie par octet

- ▶ Lecture: `int fgetc (FILE *stream);`
lit un `unsigned char` sur l'entrée standard et renvoie un `int` égal à EOF en cas d'erreur ou de fin de fichier.
- ▶ Ecriture: `fputc(int c, FILE *stream);`
écrit le caractère `c` dans le fichier manipulé avec `stream`.

Entrée sortie formatées de chaînes de caractères

- ▶ Lecture: `char * fgets (char * s, int size, FILE * flux);`
lit au plus `size - 1` caractères sur une ligne du fichier associé à `flux` et les place à l'adresse `s`. Si un retour-chariot est lu, il est mémorisé. Le caractère `'\0'` est placé à la fin de la ligne.
Renvoie `s` ou `NULL`.
- ▶ Ecriture: `int fputs (const char *s, FILE *flux);` écrit la chaîne de caractères dans le fichier associé à `flux` sans le `'\0'`.
Renvoie EOF en cas de problème.

Entrée sortie formatées

- ▶ Lecture:

`int fscanf (FILE * flux, const char * format, ...);`
effectue la lecture dans le fichier associé à `flux`. Les autres arguments sont identiques à ceux de `fscanf`.

- ▶ Ecriture:

`int fprintf (FILE *flux, const char *format, ...);`
effectue l'écriture dans le fichier associé à `flux`. Les autres arguments sont identiques à ceux de `printf`.

Il existe également des fonctions de lecture et écriture dans des chaînes de caractères:

`int sscanf(const char *str, const char *format, ...);` et
`int sprintf(char *str, const char *format, ...);`
voir man.