

Système d'exploitation

Jean-Sébastien Coron

Université du Luxembourg

November 8, 2009

- Les fichiers
 - Ouverture d'un fichier.
 - Lire et écrire des données dans un fichier.
- Exécution d'une commande UNIX en C

- Les fichiers sont identifiés par un flux qui est de type `FILE`
*
- Ouverture d'un fichier: création d'un flux:
 - `FILE *f; // déclaration du flux`
 - `f=fopen("nom.ext", "r");` en lecture ou bien:
 - `f=fopen("nom.ext", "w");` en écriture.
 - `FILE *fopen(char *file, char *mode)`
- Flux existant:
 - `stdin`: l'entrée standard: le clavier.
 - `stdout`: la sortie standard: l'écran.
 - `stderr`: la sortie des erreurs standard: l'écran.

- Écriture dans un fichier:
 - `fprintf(f, "bonjour");` écriture dans le fichier `f`.
 - `fprintf(f, "%d", i);` stocke l'entier `i` dans le fichier `f`.
- Fonctionne comme la fonction `printf`.
 - La seule différence est que l'on doit indiquer un flux.
 - `fprintf(stdout, "hello");` équivalent à `printf("hello");`

- Lecture dans un fichier:
 - `fscanf(f, "%d", &i)` ; lecture d'un entier dans un fichier.
 - `fscanf(f, "%s", s)` ; lecture d'une chaîne dans un fichier (s'arrête au premier espace).
- Fonctionne comme la fonction `scanf`.
 - La seule différence est que l'on doit indiquer un flux.
 - `fscanf(stdin, "%d", &i)` ; équivalent à `scanf("%d", &i)` ;

Exemple

```
#include <stdio.h>
int main()
{
    FILE *f1,*f2; int i,a;
    f1=fopen("toto.txt","w");
    for(i=0;i<5;i++) fprintf(f1,"%d\n",i);
    fclose(f1);
    f2=fopen("toto.txt","r");
    for(i=0;i<5;i++) {
        fscanf(f2,"%d",&a);
        printf("%d\n",a);
    }
    fclose(f2);
}
```

Manipulation de fichier

- `int fputc(char c, FILE *f)`
 - écrit le caractère `c` dans le fichier `f`, et incrémente la position.
- `int fgetc(FILE *f)`
 - lit un caractère, et incrémente la position.
 - Si fin du fichier, on obtient `EOF`.
- `int feof(FILE *f)`
 - teste si on a atteint la fin du fichier (0 pour faux, non-nul pour vrai)
- `int fclose(FILE *f)`
 - ferme le fichier

Exemple

```
#include <stdio.h>
int main(char argc, char *argv[])
{
    FILE *f;  int c;
    f=fopen(argv[1],"r");
    while(!(feof(f)))
    {
        c=fgetc(f);
        if(c!=EOF) printf("%c",c);
    }
    fclose(f);
}
```

Lancement d'une commande

- `int system(char *comm);`
 - Lance la commande contenue dans la chaîne de caractère `comm`.
 - L'exécution du programme est bloquée jusqu'à la fin de l'exécution de la commande lancée.
- Exemple:

```
#include <stdio.h>
int main()
{
    system("ls -l");
}
```