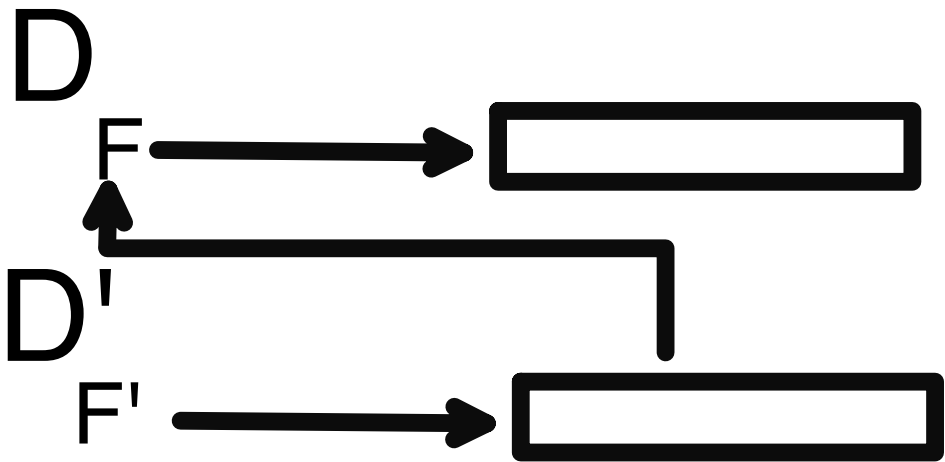


Symbolic links

Abbiamo un file, che è il primo parametro di questa funzione, possiamo creare un altro file all'interno del quale scriviamo delle informazioni (in particolare il nome) del file originale. Abbiamo un file, questo file F sta all'interno di una directory D , e quindi abbiamo un hard-link verso questo file, creiamo all'interno della directory D' un file F' che in realtà è un soft-link verso F , F' è un symbolic link. F' è un altro file con un altro I-NODE ma dentro a questo file andiamo ad inserire il nome di F .



SYNOPSIS

```
#include <unistd.h>

int symlink(const char *oldpath, const char *newpath)
```

```
int mkdir(const char *pathname, mode_t mode)
```

```
int rmdir(const char *pathname)
```

Ovviamente F può essere rimosso tranquillamente. Però abbiamo la possibilità (da un file) di identificare un altro file. Le due directory sono su due file system diversi. Quando un file è un symbolic link, ci sono applicazioni che utilizzano il contenuto di questo symbolic link per poi andare a lavorare da un'altra parte. Non appena leggo il nome su un symbolic link del file su cui devo andare a lavorare, apro l'altro file.

Creazione/rimozione di directory