



POLITECNICO
DI MILANO

INFORMATICA

Le operazioni sui file

Operazioni di base nella gestione dei file

apertura

open

memorizza sempre la posizione corrente nel file

restituisce al programma il descrittore del file

chiusura *close*

trasferisce informazioni scritte sul file

cancellazione

remove

ridenominazione

rename

per modificare il nome di un file

per spostarlo di direttorio

Con riferimento al linguaggio ^

Modalità base di accesso

complementari rispetto alle informazioni precedenti

specificate durante la fase di apertura del file

open

lettura

r read il file non può essere modificato
la posizione corrente è all'inizio del file

scrittura

W write

la posizione corrente è all'inizio del file

aggiunta

rw

a append

la posizione corrente è alla fine del file

ls-lpa

Altre modalità di accesso

Altre modalità di accesso

binary

utilizzo del file in accesso binario

+ sia scrittura che lettura

esempi

W

write in modalità testo

W+

write e read in modalità testo

rb

read in modalità **b**inaria

ab+

aggiunta **write** e **read** in modalità **b**inaria

open

parametri

nome

modalità di apertura del file

```
FILE *fp;  
char *fname = "MioFile";  
char *mode = "w";  
  
fp = fopen(fname, mode);  
/* fp == NULL se errore di apertura */
```



close

chiudere il file

```
FILE *fp;  
int result;  
  
result = fclose(fp);  
/* result != 0 se errore di chiusura */
```



quasto disco fisso



unità disco mancante

remove

parametri

nome del file

cancellare il file

```
FILE *fp;  
int result;  
char *fname = "MioFile";  
  
result = remove(fname);  
/* result != 0 se errore di cancellazione */
```

mancanza dei privilegi write

*rename*rinominare il file

```
FILE *fp;  
char *fnameOld = "VecchioNome";  
char *fnameNew = "NuovoNome";  
  
result = rename(fnameOld, fnameNew);  
/* result != 0 se errore di ridenominaz. */
```

esiste già un file con lo stesso nome

rename

rinominare il file

riposizionare

```
FILE *fp;  
char *fnameOld = "MyDir1/Vecchio";  
char *fnameNew = "MyDir2/Nuovo";  
  
result = rename(fnameOld, fnameNew);  
/* result != 0 se errore di ridenominaz. */
```

esiste già un file con lo stesso nome

non esiste la directory specificata

Gestione degli errori nell'accesso ai file

Gestione degli errori nell'accesso ai file

raggiungimento della fine del file

file end of file
feof

indica se si è raggiunta la fine del file

operazioni successive

generano messaggi di errore

verifica di eventuali errori di accesso *ferror*

clearerr cancellazione di eventuali errori di accesso

riporta al default i valori di eof e di error per il file

feof

verifica il raggiungimento della fine del file ^{*handle*}
leggendo la variabile eof associata a fp

```
FILE *fp;  
int result;  
  
result = feof(fp);  
/* result != 0 se fine file raggiunta */
```

ferror

verifica gli errori di accesso al file *handle*
leggendo la variabile error associata a fp

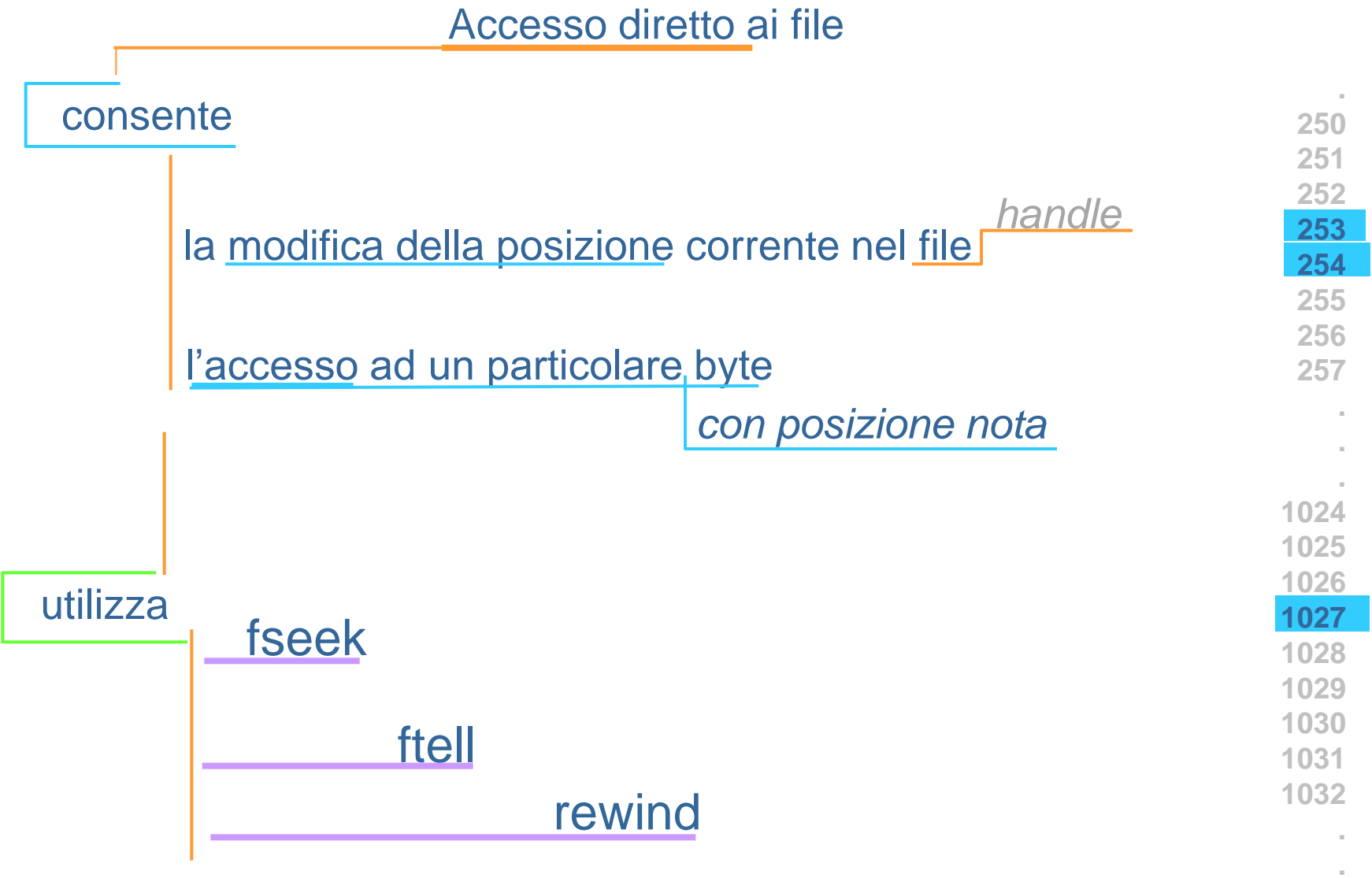
```
FILE *fp;  
int result;  
  
result = ferror(fp);  
/* result != 0 se errore di accesso al file */
```

clearerr

ristabilisce i valori di default per eof ed error sul file *handle*

```
FILE *fp;
```

```
void clearerr(fp) ;
```



Accesso diretto ai file

fseek

modifica della posizione corrente di un file *handle*

variabile long
32 bit

```
FILE *fp;
long offset;
int result, refpoint;

result = fseek(fp, offset, refpoint);
```

può assumere valori positivi o negativi
indica lo scostamento
rispetto ad un punto di riferimento
specificato dalla

se != 0 indica un errore nell'operazione

SEEK_SET inizio del file *posizione file* 125

SEEK_CUR posizione corrente *posizione del file da 1000 a* 1125

SEEK_END fine del file *posizione del file da 1000 a* 875

ftell

restituisce la posizione corrente di un ^{handle} file

```
FILE *fp;  
long offset;  
  
offset = ftell(fp);
```

rewind

riporta all'inizio la posizione corrente di un file *handle*

```
FILE *fp;
```

```
void rewind(fp);
```

```
/* equivale a fseek(fp, 0, SEEK_SET) */
```