

Apertura/chiusura di file

```
int open(char *file_name, int option_flags [, int mode])
```

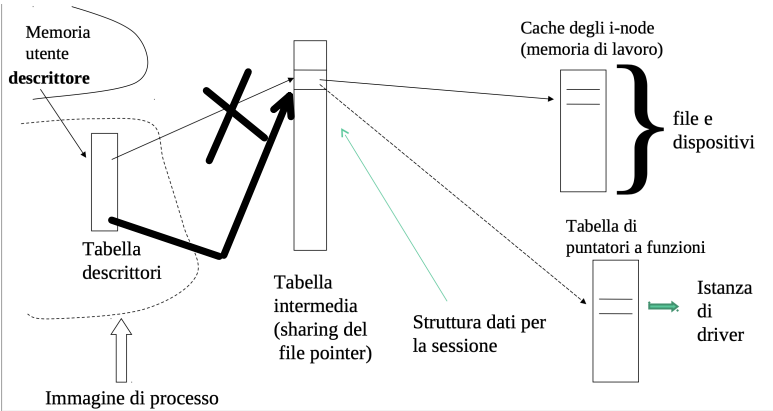
Descrizione	invoca la creazione un file
Argomenti	1) *file_name: puntatore alla stringa di caratteri che definisce il nome del file da aprire 2) option_flags: specifica la modalita' di apertura (read, write etc.) 3) mode: specifica i permessi di accesso al file in caso di creazione contestuale all'apertura
Restituzione	-1 in caso di fallimento, altrimenti un descrittore per l'accesso al file

Questa System Call, se noi passiamo un file_name, possiamo specificare una serie di altre informazioni utilizzando i parametri. Con questa open possiamo anche creare il file e contestualmente aprirlo: nel momento in cui chiamiamo questa system call, l'applicazione che chiama questa system call andrà ad occupare un i-node, verrà occupata una zona della tabella intermedia con una sessione che ci porta sull'i-node, verrà occupata una entry della tabella dei descrittori per arrivare alla sessione, a noi verrà tornato il codice numerico/descrittore per lavorare su questo canale di I/O. Possiamo chiudere il canale di I/O chiamando una close, passando il file_descriptor come input.

```
int close(int descriptor)
```

Descrizione	invoca la chiusura di un file
Argomenti	descriptor: descrittore del file da chiudere
Restituzione	-1 in caso di fallimento

Attenzione: Se io ho un canale aperto e vado a chiamare una close con un codice, e quindi arrivo sulla entry della tabella dei descrittori per chiudere il collegamento con la tabella intermedia e rilasciare la entry, NON è detto che la sessione sia rilasciata, quindi non è detto che sia rilasciato l'elemento all'interno della memoria di lavoro: questo perché qualcun altro potrebbe arrivare sulla stessa sessione con un altro canale.



La sessione rimane valida per essere usata da altri canali che possono portarci sulla sessione stessa.

O_RDONLY: apertura del file in sola lettura;
O_WRONLY: apertura del file in sola scrittura;
O_RDWR: apertura in lettura e scrittura;
O_APPEND: apertura del file con puntatore alla fine del file; ogni scrittura sul file sara' effettuata a partire dalla fine del file;
O_CREAT : crea il file con modalità d'accesso specificate da mode sole se esso stesso non esiste;
O_TRUNC : elimina il contenuto del file se esso è già esistente.
O_EXCL : (exclusive) serve a garantire che il file sia stato effettivamente creato dalla chiamata corrente.
• definiti in fcntl.h
• combinabili tramite l'operatore “ ” (OR binario)

