

Filter

Creare un programma **filter.c** in linguaggio C che accetti invocazioni sulla riga di comando del tipo:

```
filter <file.txt> <filter-1> [filter-2] [...]
```

Il programma sostanzialmente leggerà il file di testo indicato e applicherà ad ogni riga una serie di filtri indicati sulla riga di comando. Potrebbero essere presenti uno o più filtri. Il risultato finale sarà poi mostrato sullo standard output. Ogni filtro avrà la seguente struttura: - **^parola**: andrà a cercare in ogni riga le occorrenze di "parola" e le trasformerà usando solo lettere maiuscole; - ***parola**: farà lo stesso ma trasformandole usando solo lettere minuscole; - **%parola1,parola2****: andrà a cercare in ogni riga le occorrenze di "parola1" e le sostituirà con "parola2" (attenzione: le due parole potrebbero avere lunghezze diverse).

All'avvio, il programma creerà preventivamente tanti thread del tipo **Filter-n** quanti sono i filtri indicati sulla riga di comando. Tutti i thread condivideranno una struttura dati condivisa (di dimensione idonea a gestire righe lunghe al più **MAX_LEN=1024** caratteri) ed un numero idoneo di mutex e variabili condizione.

Il main thread leggerà il file indicato riga per riga; letta una riga la depositerà nella struttura dati condivisa e ne segnalerà la disponibilità al primo thread Filter-1. Il generico figlio Filter-n, letta la riga in input, applicherà la propria modifica (per tutte le occorrenze presenti) e segnalerà il completamento del proprio compito al thread seguente Filter-(n+1). L'ultimo thread del tipo Filter-n segnalerà la disponibilità della riga processata al main thread che provvederà a riversarla sullo standard output e passerà alla riga successiva.

Tutti i thread, per qualsiasi input, dovranno spontaneamente terminare e deallocare le strutture dati utilizzate alla fine dei lavori.

Tempo: 2 ore e 15 minuti

Esempio di esecuzione

```
$ ./filter letter.txt ^e _T %minix,Linux ^in %things,XYZ  
  
HEllo EvErybody out thErE usINg LINux -  
I'm doINg a (frEE) opEratINg systEm (just a hobby,  
won't BE big and profEssional likE gnu) for 386(486) At clonEs.  
this has bEEn brEWING sINcE april, and is startINg to gEt rEady.  
I'd likE any fEEedback on XYZ pEopLe like/dislike IN LINUX,  
as my OS rEsEmblEs it somEwhat (samE physical layout of thE  
file-systEm (duE to practical rEasons) among othEr XYZ).  
I've currEntly portEd bash(1.08) and gcc(1.40), and XYZ  
sEEm to work. this impliEs that I'll gEt somEthINg practical
```

withIN a fEW months, and I'd likE to know what fEaturEs most
pEople would want. Any suggEstions arE wElcomE,
but I won't promisE I'll implEmEnt thEm :-)
LINus (torvalds@kruuna.hElsINki.fi)
PS. YEs - it's frEE of any LINux codE, and it has a multi-thrEadEd fs.
It is NOT portable (usEs 386 task switchINg Etc), and it
probably nEvEr will support anythINg othEr than
At-harddisks, as that's all I havE :-(.