

Sistemi Operativi – a.a. 2017/2018

prova di laboratorio
– 19 febbraio 2018 –

Creare un programma **yet-another-fgrep.c** in linguaggio C che accetti invocazioni sulla riga di comando del tipo:

```
yet-another-fgrep [-v] [-i] [word] <file-1> [file-2] [file-3] [...]
```

Il programma emula il comportamento dell'omonimo comando di shell: legge *sequenzialmente* il contenuto dei file indicati e seleziona le righe contenenti la parola specificata; con l'opzione **v** il controllo viene invertito (vengono selezionate le righe che NON contengono la parola); con l'opzione **i** il controllo diventa *case-insensitive*; ogni riga selezionata viene riportata nello standard-output usando il nome del file di origine come prefisso.

Il processo **Father** si occuperà di creare i seguenti figli:

- un figlio **Outputer** che visualizzerà a schermo (standard output) le righe che riceve;
- un figlio **Reader** per ogni file indicato sulla riga di comando: ognuno di essi leggerà il file assegnato e ne invierà il contenuto al processo **Father**; questi figli dovranno essere creati in modo seriale (ad esempio: il secondo sarà attivato solo quando il primo avrà finito, e così via...); la lettura dei file dovrà avvenire rigorosamente attraverso la mappatura del file in memoria;

Tutti i processi comunicheranno con delle pipe dedicate. Il processo **Father** per ogni riga ricevuta applicherà il criterio di selezione specificato e manderà le righe risultanti, complete di prefisso, al figlio **Outputer**.

Tutti i processi dovranno spontaneamente terminare alla fine dei lavori. Tutte le eventuali strutture persistenti di IPC dovranno essere correttamente rilasciate in uscita.

Un esempio di output potrebbe essere il seguente:

```
$ cat nomi.txt
Marco
Alberto
Danilo
Matteo
Alfio
Vincenzo
Andrea

$ cat cognomi.txt
Mezzasalma
Facchi
Donetti
Cannella
Faro
Filetti
Cirrincione

$ ./yet-another-fgrep -i al nomi.txt cognomi.txt
nomi.txt:Alberto
nomi.txt:Alfio
cognomi.txt:Mezzasalma

$ ./yet-another-fgrep -v i cognomi.txt nomi.txt
cognomi.txt:Mezzasalma
cognomi.txt:Cannella
cognomi.txt:Faro
nomi.txt:Marco
nomi.txt:Alberto
nomi.txt:Matteo
nomi.txt:Andrea
```

Note/suggerimenti:

- per l'analisi del testo possono essere utili le funzioni di libreria `strstr()` e `strcasestr()`; quest'ultima richiede un `#define _GNU_SOURCE` prima di `#include <string.h>`

Tempo: 2 ore

Ricordarsi di inserire i propri dati (nome, cognome, matricola) nei commenti preliminari del codice sorgente.

Verrà valutata anche l'efficienza computazionale delle soluzioni algoritmiche utilizzate.

Per inviare il proprio elaborato sul server è necessario utilizzare il comando `exam-box-sync`. Verrà richiesta la password associata al proprio account e verrà data una conferma all'avvenuto caricamento. E' possibile, e fortemente consigliato, inviare il proprio elaborato più volte e periodicamente come copia di riserva (l'ambiente di lavoro degli esami risiede in memoria RAM e è pertanto di tipo non-persistente).