

## Alpha stats

Creare un programma **alpha-stats.c** in linguaggio C che accetti invocazioni sulla riga di comando del tipo

```
alpha-stats <file.txt>
```

Il programma sostanzialmente leggerà il file di testo indicato e riporterà il numero di occorrenze riscontrate per tutte le lettere dell'alfabeto inglese.

Il thread padre **P**, creerà due thread **AL** e **MZ**. Tutti i thread condivideranno una struttura dati condivisa contenente - **c**: un char/byte; - **stats**: un array di 26 unsigned long inizialmente posti a zero; - altri elementi ritenuti utili/necessari.

Per coordinarsi, i thread dovranno usare un certo numero di semafori (minimo a scelta dello studente).

Il thread **P**, dopo le operazioni preliminari sopra indicate, leggerà il contenuto del file usando la mappatura dei file in memoria (ogni altro metodo non sarà considerato corretto) e per ogni carattere alfabetico (a-z o A-Z) riscontrato provvederà a depositare in **c** la sua versione minuscola (a-z) e ad attivare, rispettivamente, il thread **AL** o **MZ** a secondo dell'intervallo di competenza. Il thread **AL** (o **MZ**) una volta attivato provvederà ad aggiornare il contatore corrispondente alla lettera ricevuta presente nell'array **stats**.

Dopo che tutte le lettere del file saranno state “analizzate” e le statistiche aggiornate, allora il thread **P** provvederà a stampare a video le statistiche presenti all'interno di **stats** e tutti i thread termineranno spontaneamente.

Tutte le strutture dati utilizzate dovranno essere ripulite correttamente all'uscita.

## Esempio di esecuzione

```
$ alpha-stats ./granni-granni.txt
```

```
Statistiche sul file: granni-granni.txt
```

```
a: 104  b: 2    c: 47   d: 24
e: 34   f: 3    g: 19   h: 19
i: 122  j: 3    k: 0    l: 10
m: 31   n: 96   o: 10   p: 28
q: 0    r: 44   s: 25   t: 43
u: 51   v: 11   w: 0    x: 0
y: 0    z: 2
```