

Esame di laboratorio di FI@BIOMED del 18 febbraio 2013

Collegatevi con la login **esame** e la passwd **Esame2012**. Cliccate sull'icona **Linux**. In tale ambiente create una directory **CognomeNome** dove salvare i vostri programmi. Aprite quindi un terminale e, usando il comando **cd**, posizionatevi all'interno della nuova directory.

Infine aprite l'editor **gedit** per creare e salvare i programmi C (nella nuova directory)

Se lo schermo si blocca usate la passwd **alpine** dell'utente **toor** per tornare a Linux.

I programmi consegnati non devono avere errori di compilazione ed devono almeno terminare. L'esame consiste di 4 passi: con Step 1 si arriva al 24 (se il programma funziona correttamente), svolgendo 2 e 3 si sale fino al 30. Il 4 è opzionale.

Step 1

Considerate un file di input "dati.txt" che contiene una sequenza di stringhe: S1 ... Sn

Esempio di file dati.txt: milano caltanissetta pisa

Scrivere un programma C che apre il file di input "dati.txt" (con il formato specificato sopra) ed inizializza una variabile globale DATI (matrice di char) con le stringhe lette dal file.

Utilizzare una variabile ausiliaria N per memorizzare il numero di stringhe lette dal file.

Il programma deve inoltre stampare su video la lista di stringhe lette partendo dall'ultima e aggiungendo dopo ogni singola parola tanti asterischi quante sono le lettere della parola stessa.

Esempio: se dati.txt contiene milano caltanissetta pisa

Il programma deve stampare : pisa **** caltanissetta***** milano*****

Step 2

Aggiungere al programma la definizione di una funzione **vocali** che prende come parametri

- una matrice bidimensionale A (compatibile con DATI)

- un numero intero N che corrisponde al num. di stringhe in A

- restituisce come valore di ritorno quante vocali diverse sono contenute nelle stringhe in A.

Nel main inserire la chiamata alla funzione **vocali** su DATI ed N e stampare il risultato su video.

Esempio: se dati.txt contiene milano caltanissetta pisa

Allora il valore stampato deve essere 4 (le vocali a,e,i,o)

Step 3

Aggiungere al programma la definizione di una procedura **numero** che prende come parametri

- una matrice bidimensionale A (compatibile con DATI)

- un numero intero N che corrisponde al num. di stringhe in A

- un vettore B di interi

- un intero F passato per riferimento

e calcola in B quante vocali (diverse) ci sono in ognuna delle stringhe in A, ed in F il valore 1 se esiste una stringa che usa tutte le vocali, il valore 0 altrimenti.

Chiamare nel main la procedura **numero** su DATI, N, su una nuova matrice B e un intero F e stampare B ed F su video.

Esempio: se dati.txt contiene milano caltanissetta pisa aiuole

Allora il programma stampa: milano 3 caltanissetta 3 pisa 2 aiuole 5 0

Step 4

Aggiungere al programma una procedura **ordina** che prende come parametri

- una matrice bidimensionale A (compatibile con DATI)

- un numero intero N che corrisponde al num. di stringhe in A

- ordina (con un algoritmo a piacere) le stringhe in A in maniera crescente in base a quante vocali (diverse) sono contenute in ognuna di esse.

Chiamare la funzione **ordina** su DATI e N e stampare su video la matrice ottenuta.

Esempio: se dati.txt contiene milano caltanissetta pisa

Allora la nuova matrice sarà pisa (2 vocali) milano (3 vocali) caltanissetta (3 vocali)