

Programmazione I
Prova di programmazione – 18 Febbraio 2021 – 2 ore

Partendo dal frammento di codice fornito, realizzare un programma per la gestione di un codice costituito da una sequenza di N righe consecutive. Ogni riga è costituita da M caratteri alfabetici maiuscoli. Né M né N sono noti a tempo di scrittura del programma. All'avvio del programma, il codice è vuoto (zero righe). Il programma fornisce le seguenti funzionalità.

1. **inizializza_codice(M, N)** Inizializza il codice a contenere N righe da M caratteri ciascuna. Legge da *stdin* il contenuto del codice, ossia i caratteri da inserire in ciascuna riga. Il precedente codice è perso.
2. **stampa_codice** Stampa il codice su una sola linea, separando le righe con il carattere |. Ad esempio, stampa col seguente formato il seguente codice da quattro righe di tre caratteri ciascuna:
P H A | U T G | H C Z | W E P
3. **[3] salva_codice** Salva il codice in un file **BINARIO** (obbligatorio per la correzione di questo punto e dei successivi) dal nome predefinito. Ricordare che salvare l'indirizzo contenuto in un puntatore ad un array non equivale a salvare l'array stesso.
4. **[3] carica_codice** Carica il codice dal file. Il precedente codice è perso.
5. **[3] ritorna_carattere(i, j)** Ritorna il carattere nella posizione i -esima della riga j -esima. Gli indici partono da zero. Ad esempio, dato il codice dell'esempio al punto 2, ritorna il carattere **C** se eseguita con $i=1$ e $j=2$.
6. **[6] ruota_codice** Trasforma il codice corrente, che ha N righe da M caratteri, in un codice da M righe ed N caratteri, in cui il carattere i -esimo della riga j -esima è uguale al carattere j -esimo della riga i -esima del codice originale. In pratica, la prima riga del nuovo codice contiene i primi caratteri delle righe del vecchio codice, la seconda riga i secondi caratteri, e così via. Ad esempio, trasforma il codice dell'esempio al punto 2, in
P U H W | H T C E | A G Z P

I parametri di ingresso delle funzionalità sono solo indicativi. Gestire opportunamente le situazioni di errore, tranne l'*overflow* e l'inserimento di dati in formato errato da *stdin*.

Per ottenere un output compatibile con il tester, per il punto 5 fare stampare dal programma il carattere ritornato, in caso di successo, oppure **FALLITO** in caso contrario. Inoltre, se prima delle righe che saranno controllate dal tester avete stampato altre righe, controllate che ci sia un accapo prima della prima riga da controllare (altrimenti tale riga risulterà appiccicata alla precedente)

REGOLE

- Si può utilizzare ogni genere di manuale e di materiale didattico
- Per superare la prova, bisogna svolgere almeno i punti 1 e 2. Se si svolgono solo tali punti, il programma deve essere perfettamente funzionante. Il voto ottenuto in questo caso è 18.
- Ciascuna funzionalità DEVE essere implementata mediante almeno una funzione.
- Il voto massimo si ottiene se
 - a) il programma è perfettamente funzionante in ogni sua parte
 - b) tutti i principi di ingegneria del codice visti nel corso sono stati applicati
 - c) sono state seguite eventuali altre indicazioni presenti nella traccia in merito al voto finale