

# Programmazione I

## Prova di programmazione – 9 febbraio 2017 – 2 ore

Partendo dal frammento di codice fornito, realizzare un programma di gestione di una pila di *blocchi di dati*. Ogni blocco è costituito da un numero variabile di valori interi. La dimensione massima di ogni blocco non è nota a tempo di scrittura del programma. La dimensione massima della pila è invece definita a tempo di scrittura del programma. All'avvio del programma la pila è vuota. Il programma deve fornire le seguenti funzionalità.

1. **inserisci\_blocco\_in\_cima(v)** Inserisce un blocco in cima alla pila. Il blocco deve contenere i valori interi presenti nel vettore **v**.
2. **stampa\_pila** Stampa il contenuto di ciascun blocco, dall'ultimo al primo, con i valori di ciascun blocco riportati su una riga. Quindi, potrebbe stampare, ad esempio, per una pila contenente tre blocchi  
2 1 10  
0 -3  
1 0 0 1
3. **elimina\_blocco\_in\_cima()** Elimina il blocco in cima alla pila. Ad esempio, se si invoca questa funzionalità ed il contenuto della pila è come nell'esempio al punto 2. il nuovo contenuto sarà:  
0 -3  
1 0 0 1
4. **salva\_stato** Salva il contenuto della pila in un file di testo dal nome definito a tempo di scrittura del programma.
5. **carica\_stato** Carica il contenuto della pila dal file. Il contenuto precedente è perso.
6. **fondi\_blocchi(M)** Fonde il numero massimo di blocchi, presenti nella pila a partire dalla cima, tali che la somma del numero di elementi che contengono è al più **M**. Ad esempio, se invocata con **M=6** sulla pila di cui all'esempio al punto 2, fonderebbe i primi due blocchi, ed il nuovo contenuto della pila diverrebbe:  
2 1 10 0 -3  
1 0 0 1

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 e di stringhe troppo lunghe da *stdin*.

---

### REGOLE

- Si può utilizzare ogni genere di manuale o di materiale didattico di altra natura
- Per superare la prova, il programma deve essere perfettamente funzionante nelle parti 1 e 2. Il voto ottenuto in questo caso è 18.
- Ciascuna funzionalità DEVE essere implementata mediante almeno una funzione.
- Il voto massimo (almeno 30) 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