

Programmazione I

Prova di Programmazione – 20 febbraio 2012 – 2 ore 10 min

Realizziamo un programma di gestione di un vettore dinamico di interi. Il vettore è caratterizzato da una *capacità*, ossia da un numero massimo possibile di elementi, anch'essa dinamica. All'avvio del programma il vettore ha capacità nulla. Partendo dal frammento di codice fornito, implementare le seguenti funzionalità.

1. **aggiungi_elemento(val)** Se il vettore ha capacità nulla, porta a 2 la capacità del vettore e vi inserisce un solo elemento di valore **val**. Se invece il vettore ha capacità maggiore di zero e c'è ancora spazio, aggiunge il valore **val** in fondo al vettore.
2. **stampa_vettore** Stampa la capacità del vettore ed i valori degli elementi del vettore, col formato mostrato nel seguente esempio. Se il vettore ha capacità 4 e contiene i valori [-5, 10, 56], stampa:
4 [-5, 10, 56]
3. **salva_vettore** Salva il vettore in un file di testo dal nome predefinito.
4. **carica_vettore** Carica il vettore dal file. Contenuto e capacità precedenti sono persi.
5. **aggiungi_elemento2(val)** Aggiunge il valore **val** in fondo al vettore. Se non c'è abbastanza spazio raddoppia la capacità del vettore prima dell'inserimento.
6. **elimina_elemento** Elimina l'elemento in fondo al vettore. Se il numero di elementi rimanenti è inferiore o uguale ad un quarto della capacità del vettore, dimezza la capacità del vettore.
7. **partiziona_vettore** Trasforma il vettore in un vettore contenente prima tutti gli elementi di valore positivo, poi quelli di valore negativo o nullo. In ciascuna delle due partizioni, non è necessario che l'ordine tra gli elementi sia lo stesso che vi era nel vettore di partenza. Ad esempio, il vettore [10, -3, 4, 5, -14, -6, 2] potrebbe divenire [10, 4, 5, 2, -14, -6, -3].
Il voto massimo si ottiene se si realizza questa funzionalità senza utilizzare alcun vettore di appoggio.

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

REGOLE

- Si può utilizzare ogni genere di manuale o di altro materiale didattico
- Per superare la prova, il programma deve essere perfettamente funzionante nelle parti 1, 2 e 3. 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