



**Laboratorio di Algoritmi e  
Strutture Dati**

Aniello Murano  
<http://people.na.infn.it/~murano/>

Murano Aniello - Lab. di ASD  
Sesta Lezione

1



**Esercitazione di  
laboratorio**

Murano Aniello - Lab. di ASD  
Sesta Lezione

2

## Esercizio

- Si implementi in linguaggio C una gestione di una coda di n elementi con due Stack.
- Suggestioni:
  - Siano H e T due Stack tali che la coda sia il risultato della concatenazione dello Stack H (partendo dal Top al Bottom) con lo Stack T (dal Bottom al Top).
  - Nella situazione iniziale, tutti gli elementi sono posti nello Stack H dove l'elemento al Top è la testa (Head) della coda, mentre quello al Bottom rappresenta la fine della coda (Tail)
  - Quando H è vuoto allora si svuota il reverse di T in H. In dettaglio, per ogni elemento di T, si farà il Pop in T e il Push in H, fino a quando T non diventa vuoto.
  - Per cancellare un elemento dalla coda, si farà un POP dallo stack H, il quale non sarà mai vuoto a meno che l'intera coda non diventi vuota.
  - Per inserire un elemento nella coda si fa un Push nello Stack T.



Murano Aniello - Lab. di ASD  
Sesta Lezione

3

## Implementazione

- Scrivere un programma in linguaggio C che implementi le funzioni Dequeue e Enqueue, viste a lezione, utilizzando due stack per la gestione della coda.
- Attenzione: L'implementazione deve avvenire modificando al minimo il main() già implementato per la gestione delle code. Inoltre, le funzioni di gestione delle code (dequeue, enqueue ecc.) devono utilizzare le funzioni di gestione degli stack indipendentemente da come sono implementati gli stack.
- Discutere sulle eventuali cambiamenti di complessità di Dequeue e Enqueue in questa nuova implementazione rispetto alle implementazioni viste nella lezione precedente.
- Domanda: Quante volte ciascun elemento sarà oggetto di un Push e di un Pop prima di lasciare la coda?



Murano Aniello - Lab. di ASD  
Sesta Lezione

4