

Esercizio 1

Una *coda* è una struttura dati con le seguenti proprietà:

- è una collezione di elementi
- mantiene l'ordine degli elementi che contiene
- è dotata di un *inizio* e di una *fine*

Le operazioni principali che posso eseguire su una coda sono:

- *Accodamento di un elemento*:
Detta anche operazione di *push*, aggiunge un elemento alla fine della coda.
- *Estrazione di un elemento*:
Detta anche operazione di *pop*, rimuove un elemento dall'inizio della coda.

In questo esercizio implementeremo una piccola libreria per gestire *code di char*. La coda è rappresentata da una *lista concatenata* di elementi di tipo

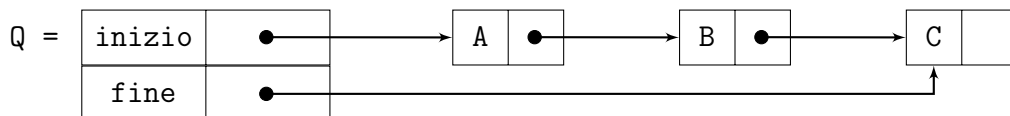
```
struct nodo {  
    char chiave;  
    nodo *next;  
};
```

La libreria definisce la struttura

```
struct coda {  
    nodo *inizio;  
    nodo *fine;  
};
```

dove i due campi puntano rispettivamente al primo e all'ultimo elemento nella coda.

Esempio. Una coda Q con tre elementi A, B, e C è rappresentata in questo modo:



La libreria deve implementare le seguenti funzioni:

- Il costruttore `coda()` che costruisce una *coda vuota* dove `inizio` e `fine` sono uguali a `NULL`.
- `void push(char c, coda &Q)` che aggiunge un nuovo elemento con `chiave == c` alla fine della coda Q;
- `char pop(coda &Q)` che estrae l'elemento all'inizio della coda Q, lo rimuove dalla coda e ne ritorna la *chiave*;
- `bool e_vuota(coda Q)` che ritorna `true` se e solo se la coda Q è vuota.

Implementare il codice delle funzioni della libreria, ed utilizzarle nel file `main.cpp` per scrivere un programma che controlla in corretto funzionamento delle operazioni di estrazione ed inserimento. Il programma legge da `cin` una sequenza di `char`:

- se è una lettera, fa il **push** della lettera nella coda;
- se è un asterisco '*' fa il **pop** dalla coda e stampa il valore estratto. Se la coda è vuota stampa "vuota".
- se è un punto '.' termina.

Esempio: dato l'input

C O D * A * * * * N E L * * M E * * * Z Z * O * * .

il programma stampa:

C O D A vuota N E L M E Z Z O

Correttezza: scrivere precondizioni e postcondizioni per tutte le funzioni definite dalla libreria, e dimostrarne la correttezza.