

Esame: PROGRAMMAZIONE E AMMINISTRAZIONE DI SISTEMA

Progetto c++

Nome: Umberto

Cognome: Cocca

Matricola: 807191

E-mail: [u.cocca@campus.unimib.it](mailto:u.cocca@campus.unimib.it)

## **CBUFFER**

Il progetto richiede di creare una classe cbuffer generica che implementa un buffer circolare di tipo T. Per fare ciò all'interno della classe cbuffer ho creato una struct nodo nel quale viene salvato il generico valore chiamato value e il puntatore alla nodo successivo, un puntatore che punta alla testa del buffer (cioè che punta alla struct più vecchia) e infine una \_size che rappresenta la capacità del buffer.

Da notare che nonostante il progetto chiedesse una capacità fissa, \_size non è un valore const, questo perché sia il costruttore copia sia l'operatore di assegnamento vanno ad accedere non solo in lettura alla \_size. A tal proposito per non rendere modificabile tale valore dall'esterno non è stato creato un metodo per settare una quantità diversa.

La funzione di inserimento di un elemento nel buffer circolare è stata implementata tenendo presente che l'ultimo nodo del buffer è sempre e comunque un nodo vuoto, ovvero un nodo con indirizzo di memoria nullo e senza valore. Tenendo presente quanto detto, l'inserimento degli elementi nel buffer avviene nella posizione del puntatore che punta a 0, ovvero all'ultimo nodo. Questo è vero finché la quantità degli elementi inseriti è inferiore alla capacità del buffer (\_size). Una volta superata la \_size allora si procede con il salvataggio del puntatore che punta all'elemento successivo alla testa, dopodiché la head si elimina e la si riassegna all'elemento che punta il puntatore precedentemente salvato. L'elemento da inserire verrà invece messo in coda. In questo modo riusciamo a ottenere l'effetto "circolare" permettendo inoltre di far coincidere l'elemento più vecchio con l'inizio del buffer.