

Relazione progetto C++ Febbraio 2022

Nome: Niccolò

Cognome: Balzarotti

Matricola: 852003

Mail : n.balzarotti2@campus.unimib.it

INTRODUZIONE

Dopo aver valutato il problema ho deciso di implementare la classe set utilizzando le linked list.

La lista contiene dei nodi, il nodo è composto da un puntatore al nodo successivo e da un puntatore all'elemento, che è costituito dal valore.

TIPI DI DATI

Ho utilizzato una struttura dati (struct): nodo.

Il nodo contiene un puntatore al nodo seguente e un puntatore al suo valore.

IMPLEMENTAZIONE E METODI IMPLEMENTATI

La classe set dispone di questi attributi:

il puntatore al nodo head (primo nodo della lista), la dimensione della lista contenente i nodi e infine il funtore di uguaglianza (_equals).

All'interno della classe set ho implementato i 4 metodi fondamentali:

Il costruttore di default, l'operatore di assegnamento, il copy constructor e infine il distruttore.

Gli ultimi due metodi utilizzano una funzione chiamata svuota che permette di svuotare completamente la lista.

Basandomi sulle richieste della traccia di esame ho deciso di implementare e utilizzare altri metodi:

Il costruttore secondario per set che crea una set riempita con dati presi da una sequenza identificata da un iteratore di inizio e uno di fine. Gli iteratori sono su un tipo diverso da quella della classe set e quindi viene effettuato un casting.

Il metodo add che permette di inserire un valore in coda alla lista senza duplicati sfruttando il funtore di uguaglianza.

Il primo valore inserito nella lista viene sempre inserito in testa quindi all'interno del nodo head mentre i valor seguenti vengono aggiunti alla lista mediante l'uso di un while.

Il metodo svuota viene utilizzato all'interno del distruttore, del copy constructor, del metodo filter_out e dei metodi operator+ e operator- della set, questo metodo permette di svuotare la set eliminando tutti i nodi presenti permettendo una corretta gestione della memoria.

Il metodo is_empty mi permette di sapere se una set è vuota oppure no.

Il metodo remove mi permette di rimuovere un nodo passato come parametro all'interno della set sfruttando il funtore di uguaglianza. Rimuovendo un nodo il nodo precedente verrà aggiornato e il suo puntatore punterà al nodo successivo del nodo eliminato.

Il metodo cerca mi permettere di cercare un nodo all'interno della set passando come parametro un valore, questo metodo viene utilizzato all'interno di altri metodi della set che sono operator- e operator==.

Il metodo operator== questo metodo mi permette di sapere se due set passate come parametri sono uguali (anche size uguale) confrontando tutti i nodi che sono al loro interno e utilizzando anche il metodo cerca.

Il metodo operator [] metodo che mi permette di estrarre dalla set un nodo che si trova in una determinata posizione che viene passata come parametro.

Il metodo filter_out metodo che permette di applicare una condizione (predicato) passato come parametro agli elementi della set passata come parametro, il metodo restituisce una set che conterrà solo gli elementi che soddisfano il predicato. In questo metodo inoltre viene utilizzato anche il metodo svuota.

Il metodo operator+ metodo che permette di unire due set passate come parametro e restituisce una set contenente gli elementi di entrambe le set senza duplicati sfruttando il metodo add per aggiungere gli elementi, inoltre in questo metodo viene utilizzato il metodo svuota.

Il metodo operator- metodo che permette di intersecare due set passate come parametro e restituisce una set contenente gli elementi comuni di entrambe le set senza duplicati sfruttando il metodo add per aggiungere gli elementi, inoltre in questo metodo viene utilizzato il metodo svuota.

Un altro metodo usato è quello per la ridefinizione dell'operatore di stream << per la set (operator<<), questo metodo permette la stampa della set.

Infine la classe set implementa un const iterator di tipo forward.

MAIN

Ho utilizzato il main per effettuare tutti i test sulla set, in particolare ho effettuato test sulla set utilizzando tipi interi, stringhe e infine tipi custom.

Il primo test che ho effettuato è stato su una set di interi nel quale ho testato tutti i metodi principali e il const iterator tramite delle assert, successivamente ho eseguito gli stessi test per il tipo stringa.

L'ultimo test effettuato è stato quello per la set di tipi custom, in particolare ho deciso di testare i metodi su set composte da Punti.

Ho implementato nel main la struct punto dopo di che ho testato tutti i metodi principali e il const iterator come per i tipi precedenti.