

Nome: Nicolò
Cognome: Ricciardo Calderaro
Matricola: 845143
E-Mail: n.ricciardocalder@campus.unimib.it

ESAME DI PROGRAMMAZIONE C++

Progetto C++

Il progetto richiedeva la realizzazione di una classe generica che implementa uno stack.

La classe in questione è stata chiamata “gstack” ovvero “generic stack”. Essa permette di creare uno stack (gestito tramite un puntatore a dati generici di tipo T); gli altri dati membro sono la grandezza dello stack e una variabile che punta alla cima dello stack. Alla sua creazione, lo stack avrà sempre grandezza uguale a zero e il dato `_top` sarà uguale a -1. `_top` assumerà il valore dell'indice dell'ultimo elemento dell'array dinamico.

Oltre ai metodi fondamentali per una classe, sono stati implementati alcuni metodi essenziali per poter gestire uno stack:

- Il metodo “push” permette di inserire dati all'interno dello stack e di porli in cima.
- Il metodo “pop” si occupa di togliere dallo stack l'elemento in cima alla pila.
- Il metodo “size” permette di conoscere la grandezza dello stack.

Sono stati implementati successivamente i metodi richiesti dalla traccia del progetto:

- Il metodo “clear” permette di eliminare tutti i dati all'interno della pila, settare la `_size` a zero e mettere `_top` uguale a -1.
- Il costruttore secondario che prende in input due iteratori generici permette di costruire un nuovo stack riempito con gli elementi interni allo stack di riferimento.
- Il metodo “rewrite” permette di riscrivere uno stack; ha la stessa funzionalità del costruttore secondario ma permette di riscrivere stack già creati. Qualora la pila fosse piena, viene prima svuotata e poi riempita con i nuovi elementi.
- Il metodo “removeif” permette di eliminare gli elementi dello stack che soddisfano un predicato generico P passato come parametro.

Dato che è stato richiesto di implementare un iteratore di sola lettura che ritorna gli elementi a partire da quello in cima alla pila, è stato implementato un “forward

iterator” dove però il metodo begin si riferisce alla cella (size – 1) e per questo sono stati anche cambiati gli operatori di pre-incremento e post-incremento in modo che decrementino il valore. Questo ha permesso di soddisfare la richiesta del progetto.

Sono stati implementati altri metodi: “is_empty” controlla che il gstack sia vuoto ritornando un valore booleano, “swap” invece permette di scambiare i dati di due gstack. Questi sono metodi ausiliari alla classe e in quanto tali sono stati messi privati. Infine vi sono il metodo “value” che permette di leggere l’elemento dello stack all’indice i-esimo e il metodo “print” che permette di stampare la pila.

Progetto Qt

Il progetto richiedeva la creazione di un editor di testo minimale.

Nell’editor di Qt creator è stato aggiunto un solo elemento: un textEdit che permette l’inserimento e cancellazione di testo.

Le successive funzionalità richieste sono state gestite tramite la toolbar che presenta due voci.

-File:

- Nuovo: permette di aprire un nuovo file. Se il file precedente presentava del testo, esso viene cancellato.
- Apri: permette di aprire i file con estensione .rtf presenti nella cartella.
- Salva: se il file non è stato salvato ancora, viene aperta una finestra che permette di dare il nome al file. Verrà aggiunta di default l’estensione .rtf; se il file era già stato salvato in precedenza, esso viene sovrascritto con le ultime modifiche apportate.
- Salva con nome: viene aperta una finestra che permette di dare il nome al file. Verrà aggiunta di default l’estensione .rtf
- Esci: permette di chiudere il programma.

-Modifica:

- Font: apre una “font dialog” che permette di cambiare la grandezza del testo ma anche di cambiare il tipo di font.
- Grassetto: permette di mettere in grassetto il testo selezionato.
- Corsivo: permette di mettere in corsivo il testo selezionato.
- Sottolineato: permette di sottolineare il testo selezionato.