«Алгоритми та структури даних»

2 курс, група 1.2

Ісаєв Михайло

Лабораторна робота №3

Варіант 13

Реалізація АТД: стек, використовуючи статичний масив.

Мова програмування: C++

*Стеком* називається послідовність елементів одного і того самого типу, до якої можна додавати нові елементи і видаляти елементи послідовності. Причому, як і додавання, так і виддалення елментів виконується з одного і того самого кінця послідовності, який називається *вершиною стека.*

Функціональний опис:

STACK[T]

Функції

push : STACK[T] x T -> STACK[T]

pop : STACK[T] -/> STACK[T]

empty : STACK[T] -> BOOLEAN

top : STACK[T] -/> T

newstack : -> STACK[T]

Аксіоми

LIFO:

top(push(s,x)) = x

pop(push(s,x)) = s

empty(newstack) = true

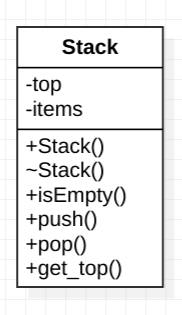
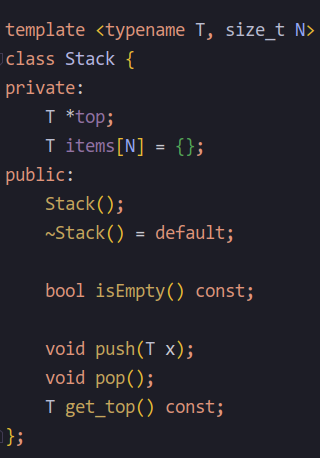
empty(push(s,x)) = false

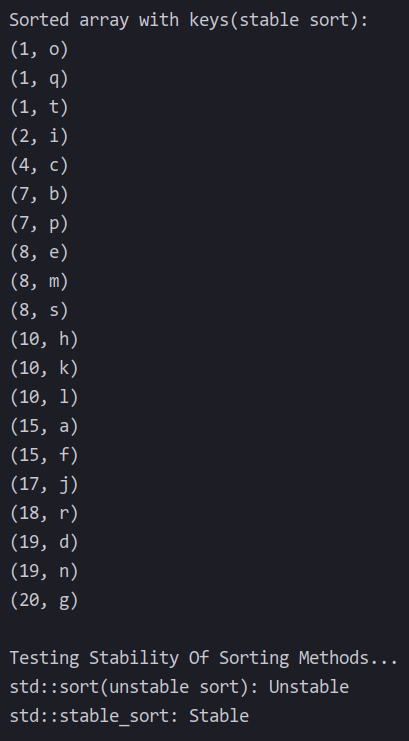
Передумови

pop (s: STACK [T]) require not empty (s)

top (s: STACK [T]) require not empty (s)

Діаграма класу, який побудований, використовуючи мову С++:



Діаграма класу *Stack* Клас *Stack* в C++

\*top - поле, вказівник для типу *T,* який вказує на верхній елемент стеку(спочатку, та коли стек порожній, вказує на елемент в пам’яті перед елементом з індексом 0 масиву *items.*

*items* - поле, статичний масив, розміру N(так як ми використовуємо статичний масив, в якому обмежується розмір, то стек *обмежений*) для сбереження елементів стеку

*Stack()* - конструктор, створює порожній стек

*~Stack() -* стандартний деструктор

isEmpty*() -* метод, повертає true, якщо стек пустий

*push(T x) -* метод, додає елемент x типу *T* до стеку

*get\_top() -* метод, повертає верхній елемент стеку

Результати тестування функцією *test\_stack:*

