**Міністерство освіти і науки України**

**Львівський національний університет імені Івана Франка**

**Факультет прикладної математики та інформатики**

Кафедра програмування

Лабораторна робота №2

**СТЕК ТА СТЕК STL**

з курсу ‘‘Алгоритми та структури даних’’

Виконав:

Студент групи ПМІ-12

Бенько Володимир Сергійович

Львів – 2023

**Стек (реалізований списком)**

Стек – лінійна, впорядкована структура даних, до якої можна доступатись лише з одного кінця (за принципом LIFO). Реалізований списком стек містить приватну структуру Item, яка містить значення та вказівник на попередній елемент стеку, вказівник на верхній елемент, розмір стеку та набір публічних методів:

* is\_empty() - повертає істину, якщо стек порожній
* size() - повертає розмір стеку
* top() - повертає значення верхнього елементу стеку, кидає underflow\_exeption якщо стек порожній
* push(T val) - додає елемент на верх стеку
* pop() - видаляє елемент з верху стеку, повертає результат в вигляді логічного значення
* clear() - видаляє всі елементи зі стеку

**Складність роботи методів:**

* is\_empty: O(1)
* size: O(1)
* top: O(1)
* push: O(1)
* pop: O(1)
* clear: O(n)

**Просторова складність стеку:** O(n)

**Стек STL**

Це стек, уже реалізований у стандартній бібліотеці шаблонів. Щоб використати його у програмі, потрібно включити <stack> через #include. У стандартного стеку також є набір методів:

* empty() – повертає істину, якщо стек порожній
* size() – повертає розмір стеку
* top() – Повертає посилання на верхній елемента стеку
* push(g) – додає елемент «g» на верх стеку
* pop() – видаляє елемент з верху стеку

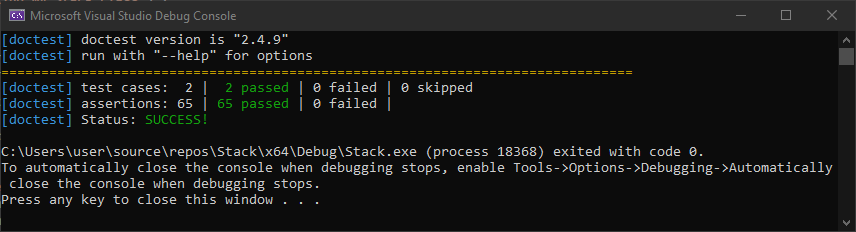
**Складність роботи методів:**

* empty: O(1)
* size: O(1)
* top: O(1)
* push: O(1)
* pop: O(1)

**Просторова складність стеку:** O(n)

**Приклад:**

Щоб переконатись, що всі методи реалізованого самостійно та стандартного стеку працюють правильно, в програмі написані юніт-тести. Усі вони проходять успішно:



**Висновок:**

Стек – зручна структура для роботи з даними, коли потрібен доступ лише до останнього елемента. Стек можна використовувати в багатьох прикладних задачах, таких як обхід графів та робота з виразами, записаними в інфіксній та постфіксній формах. Також він швидкий та займає мало пам’яті, як можна зрозуміти з часової та просторової складності. В своїх програмах можна використовувати уже реалізований стек з бібліотеки STL або один раз написати власний і користуватись ним.