**Львівський Національний Університет імені ІВАНА ФРАНКА**

Факультет прикладної математики та інформатики

**Алгоритми і Структури Даних**

**ЛАБОРАТОРНА РОБОТА №2(2025)**

**Stack та STL Stack**

Виконав:

**Заяц Ростислав Васильович**

Кафедра Програмування. Група ПМО-11с

Оцінка:

Перевірив:

**Тема.** Stack та STL Stack як LIFO структура даних.

**Мета.** Ознайомитись з Стеком, навчитись реалізовувати його сутність в С++, дослідити його роботу та навчитись використовувати. Написати тести для Стеку та СТЛ Стеку.

**Stack.**

Стек - це абстрактна структура даних, яка слідує принципу "останній прийшов - перший пішов" (LIFO - Last In, First Out). Це означає, що елемент, який був доданий останнім, буде вилучений першим.

**Реалізація.**

**Клас стек.** Як сутність використаємо стек з проштовхуванням та спеціальним символом на дні. Для того щоб реалізувати це оголосимо клас MyStack

Зображення, що містить текст, Шрифт, знімок екрана, дизайн

Автоматично згенерований опис

Клас оголосимо, щоб користувач даремно не заморочувався, що коїться в середині нашої програми, а просто спокійно користувався методами.

Отже, клас стек має 4 приватних поля:

1. Int\* arr – основа стеку, контейнер, який буде зберігати наші дані (цілі числа)
2. Int top – змінна, яка буде зберігати останній елемент стеку, тобто його вершину.
3. Const int dno – константна змінна яка зберігає наше дно. Спеціальних символ -1.
4. Size\_t maxelv – змінна, яка буде зберігати ємніть нашого стеку.

**Конструктори.**

**Зображення, що містить текст, Шрифт, знімок екрана, ряд

Автоматично згенерований опис**

**Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис**

Конструктор створює динамічний масив з розміром size + 1, де + 1 це зарезервована ділянка для нашого дна, індекс вершини = 0, тобто на момент створення вершина це наше дно. Ємність стеку = size. Деструктор очищує динамічну пам'ять.

**Методи.** Для роботи з нашим стеком оголосимо та реалізуємо потрібні нам методи:

Зображення, що містить текст, Шрифт, знімок екрана, число

Автоматично згенерований опис

1. isEmpty() – метод, який перевіряє чи порожній наш стек. Стек порожній якщо вершина = дну, отже реалізація:

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

1. isFull() – перевірка чи стек заповнений. Стек заповнений, якщо індекс вершини = ємності стеку. Реалізація:

Зображення, що містить текст, Шрифт, знімок екрана

Автоматично згенерований опис

1. push(x) – метод, який просуває в нашому стеку елемент х. Елемент можна помістити в стек, якщо він не повний. Зображення, що містить текст, знімок екрана, Шрифт, ряд

   Автоматично згенерований опис

Якщо він не переповнений, тоді спочатку додаємо 1 до індексу вершини, потім вносимо на це місце елемент x.

1. Pop() – метод який видає нам останній елемент, що лежить у вершині. Отримати елемент можна тоді й лише тоді, коли стек не порожній, отже: Зображення, що містить текст, знімок екрана, Шрифт

   Автоматично згенерований опис

Запам’ятовуємо значення, яке лежить в вершині в додаткову змінну, занулюємо вершину, зменшуємо індекс вершини на 1. Якщо стек порожній повертаємо наше дно.

1. GetOccupancy & GetCapacity – методи, які показують нам ємніть та заповненість стеку.

Зображення, що містить текст, Шрифт, знімок екрана

Автоматично згенерований опис ємність це наш maxelv, а заповненість це індекс вершини(Одночасно і к-сть елементів, яку ми додали)

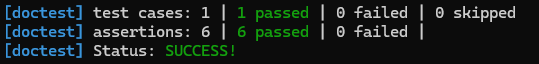
**Тести**

Тести я писав за допомогою біблоіотеки doctest, що знаходиться у відкритому доступі на гітхабі.

**Test #1 Перевірка чи правильно створено масив. Перевірка чи правильно працює GetOccupancy & GetCapacity**Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

**Result:**



**Test #2 Перевірка чи правильно заповнюється та опустошується стек.**

**Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис**

**Result:**

**Зображення, що містить текст, Шрифт, знімок екрана, ряд

Автоматично згенерований опис**

**Test #3,4 Спроба проштовхнути елемент в переповнений стек, та витягнути з пустого стеку.**

**Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис**

**Result:**

**Зображення, що містить текст, Шрифт, знімок екрана, ряд

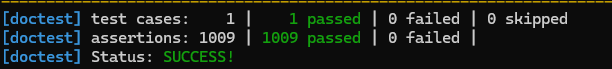
Автоматично згенерований опис**

**Test #STL Stack Загальні тести СТЛ Стеку**

**Зображення, що містить текст, знімок екрана, програмне забезпечення, Шрифт

Автоматично згенерований опис**

**Result:**

****

**Висновок.**

У мене вийшло реалізувати стек з проштовхуванням і спеціальним символом на дні на мові С++. Я навчився користуватись ним, зрозумів як він працює, а також в загальному усвідомив як працює LIFO