МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Нижегородский государственный университет им. Н.И. Лобачевского»**

**Национальный исследовательский университет**

**Институт информационных технологий, математики и механики**

**Кафедра математического обеспечения и суперкомпьютерных технологий**

**Отчет по учебной практике**

**«Структура хранения данных: Стек на массиве»**

**Выполнил:** студент группы 381706-1

Кольтюшкина Янина Вадимовна

\_\_\_\_\_\_\_\_\_\_\_ Подпись

**Научный руководитель:**

ассистент каф. МОСТ ИИТММ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Лебедев И.Г

Нижний Новгород

2018.

Содержание

[1. Введение 3](#_Toc533112702)

[2. Постановка задачи 4](#_Toc533112703)

[3. Руководство пользователя 5](#_Toc533112704)

[4. Руководство программиста 6](#_Toc533112705)

[4.1. Описание структуры программы 6](#_Toc533112706)

[4.2. Описание структур данных 6](#_Toc533112707)

[4.3. Описание алгоритмов 7](#_Toc533112708)

[5. Заключение 9](#_Toc533112709)

# Введение

Перед тем, как приступить к выполнению лабораторной работы необходимо выяснить, что такое стек в целом и стек на массиве, как один из способов реализации стека.

Стек на массиве — структура данных, представляющая собой упорядоченный набор элементов, фиксированного количества, организованных по принципу LIFO (англ. last in — first out, «последним пришёл — первым вышел»). Иными словами, доступ есть только к элементу, который был добавлен последним.

Стек это не абстрактное понятие и мы часто сталкиваемся с ним в жизни. К примеру стопка тетрадей на столе учителя. Какой бы большой она ни была, учитель первой начнет проверять ту тетрадку, которую сдали последней, ведь именно она является вершиной данного стека. И если кто-то решит сдать еще одну тетрадку, то положить ее придется на самый верх.

Но это далеко не единственный пример. Стек является одной из базовых структур данных и используется не только в программировании, но и в схемотехнике, и просто в производстве, для реализации технологических процессов и т.д.; стек используется в качестве вспомогательной структуры данных во многих алгоритмах и в других более сложных структурах.

Выражаясь более строгими терминами, в стеке мы не можем получить доступ к произвольному элементу. Мы можем только добавлять или удалять элементы с помощью специальных методов. Кроме того, у стека нет итератора.

Основная цель данной работы – разработать структуру хранения стека на массиве.

# Постановка задачи

Выполнение работы предполагает решение следующих задач:

1. Разработка и реализация класса стека - TStack.
2. Пример программы, демонстрирующая работу класса TStack.
3. Написание набора автоматических тестов с использованием Google C++ Testing Framework и проверка работоспособности методов классов.

# Руководство пользователя

При запуске программы пользователя просят ввести размер стека.



Рисунок 1 Ввод длины стека

После чего, стек заполняется значениями от 0 до введенной длины – 1. Как только он полностью заполнится, программа выведет его на экран. После чего скопирует в новый стек, выведет его на экран и сравнит с исходным.

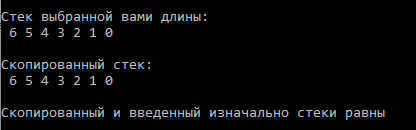


Рисунок 2 Вывод заполненного стека и его копии, равной исходному

Далее, программа заберет из изначального стека один элемент и выведет то, что там осталось. И снова сравнит с скопированным от изначального.

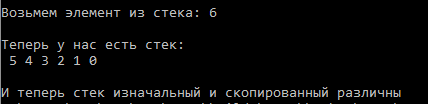


Рисунок 3 Изъятие элемента из стека, печать этого стека и сравнение с копией

# Руководство программиста

## Описание структуры программы

Программа состоит из следующих модулей:

* Модуль StackLib. Статическая библиотека. Включает в себя заголовочный файл Stack.h, в котором описаны методы с реализацией шаблонного класса стек TStack.
* Модуль StackTest. Набор тестов для класса TStack. Включает в себя файл *StackTest.cpp.* Разработаны они с помощью использования Google C++ Testing Framework.
* Модуль Stack. Пример использования стека. Включает в себя файл с реализацией *main\_Stack.cpp.*

## Описание структур данных

*Класс TStack:*

Поля:

int leng; - длина стека

T\* elem; - элементы стека

int top; - вершина стека

Конструкторы и деструктор:

TStack(int len = 0);

TStack(TStack &St);

~TStack();

Методы:

void Put(T el); - положить элемент

T Get(); - взять элемент

int GetLength(); - получить длину стека

bool IsFull(); - проверка на полноту

bool IsEmpty(); - проверка на пустоту

void PrintStack(); - вывод стека на экран

Перегрузки:

TStack& operator=(const TStack<T>& stack); - присваивание стека

int operator==(const TStack<T>& stack) const; - проверка на равенство

int operator!=(const TStack<T>& stack) const; - проверка на неравенство

## Описание алгоритмов

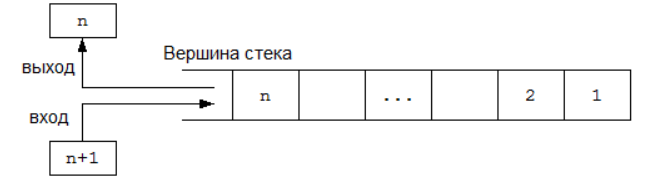


Рисунок 4 Общий принцип работы

Добавление элемента в стек:

Добавляя элемент в стек, размещаем его в первую свободную ячейку (если такая имеется) массива, выделенного под хранение данных. На эту ячейку указывает *top* – вершина стека. Затем увеличиваем значение *top* на 1.

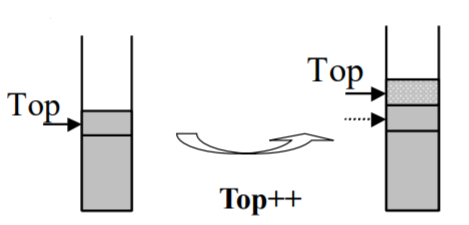


Рисунок 5 Добавление элемента

Удаление элемента из стека:

При удалении элемента из стека (если в нем есть элементы) уменьшаем значение *top* на 1 – спускаемся к последнему существующему элементу и возвращаем его значение. Само значение не затирается, т.к. оно будет перезаписано при следующем добавлении другого элемента.

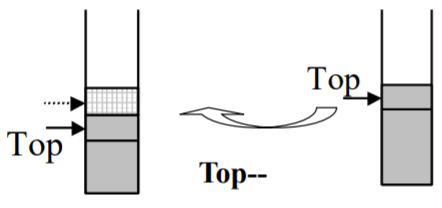


Рисунок 6 Удаление элемента

# Заключение

В этой лабораторной работе я смогла справиться со всеми поставленными задачами. Разобравшись с тем, что вообще такое стек, я реализовала класс для хранения стека на массиве TStack. Для этого была разработана библиотека StackLib, в которой полностью описан данный шаблонный класс. В отличие от предыдущих работ, здесь все тесты уже пришлось писать полностью с нуля вручную. Что и было мной удачно проделано, ведь все тесты проходятся. И, конечно, пример реализации так же полностью функционирует.

1. **Литература**
2. Гергель В.П. Методические материалы по курсу «Методы программирования 2»: [<http://www.itmm.unn.ru/files/2018/10/Primer-1.3.-Struktury-hraneniya-steka.pdf>], 2015.
3. Википедия: свободная электронная энциклопедия: на русском языке: https://ru.wikipedia.org/wiki/Стек
4. Полный курс лекций по си для студентов. Примеры, код на си, задачи на си, теория си, динамические структуры данных на си, алгоритмы на си.

https://learnc.info/about.html