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 списках»**

**Выполнил:** студент группы 381706-1

Суслов Егор Игоревич

\_\_\_\_\_\_\_\_\_\_\_ Подпись

**Научный руководитель:**

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

Нижний Новгород

2018.

Содержание

[1. Введение 3](#_Toc533083472)

[2. Постановка задачи 4](#_Toc533083473)

[3. Руководство пользователя 5](#_Toc533083474)

[4. Руководство программиста 6](#_Toc533083475)

[4.1. Описание структуры программы 6](#_Toc533083476)

[4.2. Описание структур данных 6](#_Toc533083477)

[4.3. Описание алгоритмов 7](#_Toc533083478)

[5. Заключение 8](#_Toc533083479)

[6. Литература 9](#_Toc533083480)

# Введение

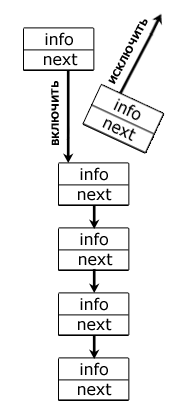
Стек характерен тем, что получить доступ к его элементам можно лишь с одного конца, называемого вершиной стека; иначе говоря: стек – структура данных типа «список», функционирующая по принципу LIFO (last in — first out, «последним пришёл — первым вышел»).

Графически его удобно изобразить в виде вертикального списка (см. рис.), например, стопки книг, где чтобы воспользоваться одной из них, и не нарушить установленный порядок, нужно поднять все те книги, что лежат выше нее, а положить книгу можно лишь поверх всех остальных.

Впервые стек был предложен в 1946 году Аланом Тьюрингом, как средство возвращения из подпрограмм. В 1955 году немцы Клаус Самельсон и Фридрих Бауэр из Технического университета Мюнхена использовали стек для перевода языков программирования и запатентовали идею в 1957 году. Но международное признание пришло к ним лишь в 1988 году.

Стек, чаще всего, реализуется на основе обычных массивов, односвязных и двусвязных списков. В зависимости от конкретных условий, выбирается одна из этих структур данных.

На рисунке показан стек, операции над элементами которого, происходят строго с одного конца: для включения нужного элемента в n-ую ячейку, необходимо сдвинуть n-1 элементов, и исключить тот элемент, который занимает n-ую позицию.



# Постановка задачи

В рамках лабораторной работы ставится задача эффективной реализации структуры данных – стека на списках.

Для работы со стеком необходимо реализовать операции:

* добавление элемента;
* удаление элемента;
* чтение верхнего элемента.

Программное решение будет выглядеть следующим образом:

1. Класс стека на списках – TStackList..
2. Класс для обработки исключений – MyException, которые могут возникнуть при выполнении различных операций.
3. Программа, демонстрирующая работу класса TStackList (получается из файла кода «StackList\_main.cpp».
4. Набор автоматических тестов с использованием Google C++ Testing Framework.

# Руководство пользователя

1. При запуске программы c пользователя спрашивается максимальный размер стека
2. Создается стек размером и автоматически заполняется числами от 1 до
3. Выводится полученный заполненный стек
4. Элементы изымаются из стека
5. Сообщение «Стек пуст»
6. Конец работы консольного приложения

# Руководство программиста

## Описание структуры программы

Программа состоит из следующих модулей:

* Модуль StackList. Содержит пример использования стека. Реализация в файле «StackList\_main.cpp».
* Модуль StackListLib – статическая библиотека. Содержит файл «StackList.h», в котором описан интерфейс и реализация шаблонного класса TStackList(2 конструктора, деструктор и 7 методов).
* Модуль StackListTest. Содержит 14 тестов, описанных в файле «StackTest.cpp»и разработанных с помощью использования Google C++ Testing Framework.
* Модуль MyExceptionLib – библиотека, содержащая класс исключений.

## Описание структур данных

#### Класс TStackList

Класс TStackList является шаблонным классом. Наследуется от класса TList со спецификатором public.

В классе TStackList определено одно поле, со спецификатором доступа protected, int size – максимальный размер стека.

Далее в публичной зоне (public):

1. Два конструктора и деструктор:

* TStackList(int \_size = 10) – конструктор с одним параметром.
* TStackList(TStackList<T> &A) – конструктор копирования.
* ~TStackList() – деструктор.

1. Методы для работы со стеком на списках:

* void Put(T A) – добавить новый элемент A в начало списка.
* T Get() – изъять с удалением элемент находящийся в начале списка.
* void Print() – вывод элементов стека на списках на экран.
* bool IsFull() – проверка стека на списках на полноту.
* bool IsEmpty() – проверка стека на списках на пустоту.
* int GetSize() – получить текущее количество элементов в стеке на списках.
* int GetMaxSize() – получить максимальное количество элементов в стеке.

## Описание алгоритмов

**Добавление звена списка в начало и в конец.**

При добавлении элемента в стек, он размещается на вершине стека. При изъятии элемента из стека берется элемент, находящийся на вершине стека. Поскольку в данной работе стеки хранится на списках, то целесообразно вершиной стека считать первый элемент списка. То есть добавление и изъятие элементов происходит только из начала списка. В этом случае сложность этих операций составит О(1). Иначе, для добавления или изъятия элементов из конца списка, пришлось бы обходить весь список поэлементно, и сложность операций составила O(n).

# Заключение

Все поставленные задачи были выполнены это подтверждается Google C++ Testing Framework (все 22 теста на класс TStackList успешно выполняются) и работает файл кода «StackList\_main.cpp» использующий наш класс.

# Литература

* Книги:

1. A.O. Грудзинский. Методы программирования, Издательство Нижегородского госуниверситета, 2006.
2. Топп У., Форд У. Структуры данных в С++. - М. Бином, 1999.
3. Мейн М., Савитч У. Структуры данных и другие объекты в С++. - М.: Издательский дом “Вильямс”, 2003.
4. Гергель В.П. Методические материалы по курсу «Методы программирования 2», Нижний Новгород, 2015.

* Ссылки в Internet:

1. Учебно-методическое пособие из электронной библиотеки ННГУ: «ЛАБОРАТОРНЫЙ ПРАКТИКУМ» : [http://www.unn.ru/books/met\_files/Pract\_ADS.pdf].