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**

**ЛАБОРАТОРНАЯ РАБОТА**

**«Структура хранения данных: Мультистек»**

**Выполнил:** студент группы 381706-2

Паузин Леонид Павлович

\_\_\_\_\_\_\_\_\_\_\_ Подпись

**Руководитель:**

Ассистент кафедры МОСТ

Лебедев Илья Геннадьевич

\_\_\_\_\_\_\_\_\_\_\_ Подпись

Нижний Новгород

2018

Содержание

[1. Введение 3](#_Toc8273512)

[2. Постановка задачи 4](#_Toc8273513)

[3.Руководство пользователя 5](#_Toc8273514)

[4.Руководство программиста 6](#_Toc8273515)

[4.1 Описание структуры программы 6](#_Toc8273516)

[4.2 Описание структур данных 6](#_Toc8273517)

[4.3 Описание алгоритмов 6](#_Toc8273518)

[5.Заключение 8](#_Toc8273519)

[6.Литература 9](#_Toc8273520)

# 1. Введение

Лабораторная работа направлена на практическое освоение динамической структуры данных мультистек.

Стек (англ. stack ‒ стопка) ‒ абстрактный тип данных, представляющий собой список элементов, организованных по принципу LIFO (англ. last in ‒ first out, «последним пришёл ‒ первым вышел»). Это значит, что мы будем иметь доступ только к последнему добавленному элементу.

**Мультистек** — структура данных, представляющая собой упорядоченный набор N стеков, фиксированного размера. Разработка класса мультистек обеспечит работу с несколькими стеками

# 2. Постановка задачи

Выполнение работы предполагает решение следующих задач:

* Разработка и реализация вспомогательного класса стека.
* Разработка и реализация класса мультистека.
* Разработка программы, демонстрирующей работу классa мультистек.

# 3.Руководство пользователя

В мультистек поочередно загружаются или удаляются элементы до тех пор, пока их число не достигнет 50. После каждого действия на экран выводится мультистек, число элементов и число перепаковок.

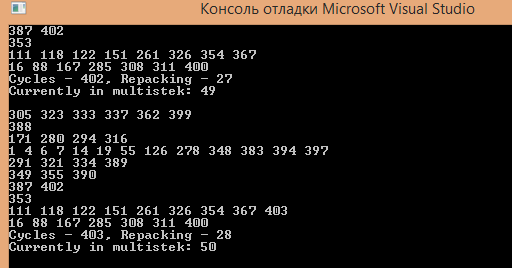


Рисунок 1. Демонстрация работы класса мультистек

# 4.Руководство программиста

## 4.1 Описание структуры программы

* multistacklib(MStack.h, TMultiStack.h) – модуль, реализующий класс мультистек
* multistack(main.cpp) – модуль, реализующий тестирование мультистека
* multistacktest(main.cpp) ‒ модуль тестирования при помощи Google C++ Testing Framework

## 4.2 Описание структур данных

TMultiStack:

Поля:

MStack <ValType> \*\* pStack – стеки

int memSize– всего выделено памяти

int currentCount – число элементов в мультистеке в данный момент

int stackCount – число стеков

ValType\* stackMem – память под стек

Методы:

int IsFull() ‒ контроль переполнения

int IsEmpty() ‒ контроль пустоты

void Put(int ns, const ValType &Val) ‒ добавить значение

ValType Get(int ns) ‒ извлечь значение

void Print() ‒ печать значений стека

## 4.3 Описание алгоритмов

Перераспределение памяти между стеками (перепаковка):

* Метод принимает целочисленный параметр – номер стека в котором не хватает памяти;
* Создаются переменные для циклов и сообщения прошло ли перераспределение памяти успешно, считается количество свободной памяти;
* Если есть свободная память, то указатели на начало стеков смещаются с учетом занятой памяти и остатка незанятой памяти, разделенной на количество стеков, нужному стеку присваивается и остаток от деления;
* Далее смещается память, которая уже была занята в стеках, причем с конца, чтобы не произошло «утечки».
* Стекам присваивается новое значение длины.

Возвращается результат перераспределения.

# 5.Заключение

В ходе выполнения лабораторной был произведен анализ задачи - установлено понятие мультистека. Была разработана библиотека MultiStackLib, содержащая шаблонный класс мультистека TMultiStack и вспомогательный класс MStack. В библиотеке реализованы методы работы с мультистеком.

Разработаны и доведены до успешного выполнения тесты, разработанные для данного программного проекта с использованием Google C++ Testing Framework.

# 6.Литература

1. Гергель В.П. Методические материалы по курсу «Методы программирования 2», Нижний Новгород, 2015.
2. Википедия: свободная электронная энциклопедия ‒ Стек: <https://ru.wikipedia.org/wiki/Стек>
3. Статья «Стек» на сайте: <https://prog-cpp.ru/data-stack/>