Министерство образования и науки Российской Федерации

Нижегородский государственный университет им. Н.И. Лобачевского

Институт информационных технологий, математики и механики

**Отчет по лабораторной работе**

**Разработка структуры данных стек и применение её для решения практической задачи**

**Выполнил**:студент группы 381606-2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тимакин Н.Е.

Подпись

**Проверил**: к.ф.-м.н., доц.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Баркалов К.А.

Подпись

Нижний Новгород 2017

**Введение**

Стек – это абстрактный тип данных, представляющий собой список элементов и организованный по принципу «последний пришёл – первым ушёл». Стек не только часто встречается в повседневной жизни (стопка тарелок или книг), но ещё с помощью него решаются многие прикладные задачи, например в теории графов. Одна из таких задач и будет рассмотрена как пример использования этого типа данных.

**Постановка задачи**

Реализовать тип данных стек как шаблонный класс, написать к нему тесты, подтверждающие его работоспособность, и тестовую программу – калькулятор, в двух вариантах: в виде консольного приложения и с графическим интерфейсом.

**Описание структуры программы**

Сам стек будет реализован как класс TStack в одном .h-файле, и для калькулятора понадобится написать отдельный класс TCalculator, который будет проводить все нужные операции для вычисления результата и использовать для этого поля типа TStack.

Программа содержит в себе 3 проекта:

1. Stack-lab – реализация классов TStack и TCalculator, тестовая программа (консольное приложение)
   * stack.h – объявление и реализация класса TStack
   * TCalculator.h – объявление класса TCalculator
   * TCalculator.cpp – реализация класса TCalculator
   * main.cpp – реализация тестовой программы
2. Test – тесты для стека и калькулятора
   * gtest-all.cc – библиотека с реализацией тестов
   * TCalculator.cpp (добавлен для работы с TCalculator)
   * test\_calculator.cpp – тесты для класса TCalculator
   * test\_main.cpp – файл, запускающий все тесты
   * test\_stack.cpp – тесты для класса TStack
3. Visual calculator – тестовая программа (приложение с графическим интерфейсом)
   * MyForm.h – объявление и реализация формы для графического приложения
   * MyForm.cpp – файл для запуска приложения
   * TCalculator.cpp (добавлен для работы с TCalculator)

Т - шаблон

**Класс TStack**

Поля:

T\* arr - память для хранения элементов

int size - текущий размер

int maxsize - максимальный размер

Методы:

TStack(int \_maxsize = 10) – конструктор по умолчанию

TStack(const TStack& st) – конструктор копирования

~TStack() - деструктор

TStack& operator=(const TStack& st) – перегрузка оператора присваивания

int isfull() – проверка на полноту

int isempty() – проверка на пустоту

T top() – возвращение вершины стека

T pop() – удаление элемента

void push(const T& el) – добавление элемента

int getsize() – возвращение текущего размера

int getmaxsize() – возвращение максимального размера

int operator==(const TStack& st) const – перегрузка операции сравнения

int operator!=(const TStack& st) const – перегрузка операции не равно

void clear() – очистка стека

**Класс TCalculator**

Поля:

string infix – инфиксная строка

string postfix – постфиксная строка

TStack<char> stc – стек для знаков

TStack<double> StD – стек для чисел

Методы:

TCalculator(string \_infix = "") – конструктор по умолчанию

void setinfix(string \_infix) – задание инфиксной строки

string getinfix() – получение инфиксной строки

string getpostfix() – получение постфиксной строки

int check() – проверка на корректность инфиксной строки

int priority(char sym) - определение приоритета символа

void topostfix() – перевод инфиксной строки в постфиксную

double calc() – подсчёт результата

**Описание алгоритмов**

**Результаты**

**Вывод**

**Литература**

Брайан Керниган, Деннис Ритчи «Язык программирования Си»

Брюс Эккель «Философия С++. Введение в стандартный С++»

Стивен Прата «Язык программирования С++. Лекции и упражнения»

**Приложение**