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**

**«Преобразование арифметических операций в обратную польскую запись»**

**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](#_Toc1154284)

[2. Постановка задачи 4](#_Toc1154285)

[3. Руководство пользователя 5](#_Toc1154286)

[4. Руководство программиста 6](#_Toc1154287)

[4.1. Описание структуры программы 6](#_Toc1154288)

[4.2. Описание структур данных 6](#_Toc1154289)

[4.3. Описание алгоритмов 7](#_Toc1154290)

[5. Заключение 9](#_Toc1154291)

# Введение

**Целью** данной лабораторной работы является практическая реализация системы, способной переводить заданные арифметические выражения в обратную польскую запись и производить с их помощью необходимые вычисления.

Ознакомимся со следующими базовыми понятиями.

**Обратная польская запись** – форма записи математических и логических выражений, при которой операнды расположены перед знаками операций. Такая форма позволяет записывать любое численное выражение без использования скобок.

**Инфиксная нотация** – обычные математические операнды, записываемые в виде: x + y. Такая форма записи привычна и удобна для людей, но не для машины. Любая подобная операция сводится к функции от двух параметров – x и y. Главными проблемами такого подхода являются: приоритет операций и ассоциативность операций. Из-за этого определение функции инфикса становится нетривиальной задачей. Зачастую, вычисление инфиксного выражения требует изменения порядка операторов и их операндов. Для наглядности и с целью банально не запутаться в выражениях, приходится использовать скобки.

**Постфиксная нотация**, именуемая также обратной польской нотацией, предлагает записывать оператор после его операнда. Так выражение xy+ эквивалентно x+y.



Рисунок 1 Постфиксная форма

У такой формы записи есть несколько неоспоримых достоинств. Первое из них – она обобщается n-адическим оператором (в то время как, инфиксная нотация на самом деле работает только с двумя операндами, т.е. по своей природе подходит только для бинарных операций). Второе - приоритетом операций служит порядок их появления, при этом никогда не понадобятся скобки.

# Постановка задачи

Одной из главных задач данной лабораторной работы является разработка эффективной системы для преобразования арифметических выражений в обратную польскую запись и нахождение результата такого выражения.

Программное решение будет выглядеть следующим образом:

1. Разработка и реализация метода перевода выражений из инфиксной в постфиксную нотацию
2. Разработка и реализация метода, вычисляющего результат выражения, записанного в постфиксной нотации.
3. Пример программы, демонстрирующий работу этих двух методов.
4. Написание набора автоматических тестов с использованием Google C++ Testing Framework и проверка работоспособности методов классов.
5. Класс для обработки исключений – MyException.

# Руководство пользователя

При запуске программы пользователя кратко информируют о том, что будет происходить далее и просят ввести математическое выражение. Далее оно еще раз дублируется на экране и переводится в обратную польскую запись. Это упрощает проверку правильности вычислений. Переведенное выражение тут же подсчитывается, а результат выводится на экран.

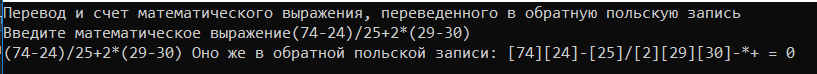


Рисунок 2 Вычисление значения выражения, переведенного в обратную польскую запись

# Руководство программиста

## Описание структуры программы

Программа состоит из следующих модулей:

* Модуль Polish. Содержит пример вычисления арифметического выражения с помощью перевода в обратную польскую запись. Реализация в файле *Polish\_main.cpp*
* Модуль PolishLib - статическая библиотека. Содержит заголовочный файл Polish.h, в котором описаны все необходимые методы для работы с обратной польской записью. Реализованы все они в исполняемом файле Polish.cpp*.* Файлы Stroka.h и Stroke.cpp, содержат описание интерфейса и реализацию вспомогательного класса *TString.*
* Модуль PolishTest. Содержит 23 теста, описанных в файле *PolishTest.cpp* иразработанных с помощью использования Google C++ Testing Framework.
* Модуль ExceptionLib – библиотека, позволяющая создавать собственные исключения.

## Описание структур данных

*Класс TString:*

Поля:

char\* strg; - массив для хранения строки

int leng; - длина строки

Конструкторы и деструктор:

TString();

TString(char\* \_strg);

TString(TString &str);

~TString();

Методы:

int GetLeng(); - получить длину строки

void DeleteStrg(); - удалить строку

Перегрузки:

TString operator + (TString &str);

TString& operator = (TString &str);

Char& operator [] (int i);

friend istream& operator >> (istream &istr, TString &str);

friend ostream& operator << (ostream &ostr, TString &str);

*Файл Polish.h:*

Методы:

int GetPriority(const char prior); - вычисление приоритета операции

TQueue<char> ConvertToPol(TString str); - преобразует выражение из инфиксной формы в префиксную

double Res(TQueue<char> que); вычисление результата операции, записанного в обратной польской нотации

bool IsPolish(char s)

## Описание алгоритмов

**Преобразование выражения из инфиксной формы в обратную польскую запись.**

* + - 1. Однозначно определим приоритет операций:

|  |  |
| --- | --- |
|  |  |

* + - 1. Для хранения операций используется стек st, для хранения выражения в обратной польской записи используется очередь que.
      2. Исходное выражение просматривается слева направо.
      3. Операнды по мере их появления помещаются в очередь que.
      4. Символы операций и левые скобки помещаются в стек st.
      5. При появлении правой скобки последовательно изымаются элементы из стека st и переносятся в очередь que. Данные действия продолжаются либо до опустошения стека st, либо до попадания в стеке st на левую скобку.
      6. Если текущая операция op, выделенная при обходе выражения, имеет больший вес, чем операция на вершине стека st, то она кладется в стек st. Иначе последовательно изымаются элементы из стека st и переносятся в очередь que. Данные действия продолжаются либо до опустошения стека st либо до того, как приоритет операции вершины стека будет не больше приоритета текущей операции op. Как только эти действия прекращаются текущая операция op кладется в стек st.

# Заключение

В этой лабораторной работе я смогла справиться со всеми поставленными задачами. Разобравшись в том, что такое обратная польская запись и по какому принципу возможно осуществить перевод арифметического выражения и найти его значение, я реализовала класс TPolish. Для этого была реализована библиотека PolishlLib, в которой полностью описаны методы, необходимые для работы с обратной польской записью. Также так описан вспомогательный класс TString. Написанный мной набор автоматических тестов, проверяет работоспособность всех реализованных методов и делает это весьма успешно. Пример использования данных методов для пользователя полностью функционирует и готов к использованию.

1. **Литература**
2. Fb.ru: Обратная польская запись: алгоритм, методы и примеры http://fb.ru/article/321181/obratnaya-polskaya-zapis-algoritm-metodyi-i-primeryi.
3. Википедия: свободная электронная энциклопедия: на русском языке: [https://ru.wikipedia.org/wiki/Обратная-польская\_запись