

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра Информационные Системы**

**ОТЧЕТ**  
**по лабораторной работе №1**  
**по дисциплине «Алгоритмы и структуры данных»**  
**Тема: Списочные структуры**

Студентка гр. 1373

\_\_\_\_\_

Жмуренко В.К.

Преподаватель

\_\_\_\_\_

Бондаренко Б.Е.

Санкт-Петербург

2022

## Цель работы:

Реализовать следующие структуры: односвязный список, динамический массив и стек. Стек можно реализовать как на базе списка, так и отдельно. Использовать стек для реализации алгоритма сортировочной станции. Разрешённые символы в исходном выражении: +, -, \*, /, ^, sin, cos, (, ), 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Для упрощения разбиения входной строки на токены разрешается отделять каждый символ пробелом.

## Основные теоретические положения:

**Стек** — абстрактный тип данных, представляющий собой список элементов, организованных по принципу LIFO (last in — first out, «последним пришёл — первым вышел»).

**Динамический массив** — массив, размер которого может изменяться во время исполнения программы за счет динамического выделения памяти.

**Односвязный список** — это динамическая структура данных, состоящая из узлов. Каждый узел будет иметь какое-то значение и указатель на следующий узел.

**Алгоритм сортировочной станции** — способ разбора математических и/или логических выражений, представленных в инфиксной нотации. Может быть использован для получения вывода в виде обратной польской нотации

## Ход работы:

Реализовал стек и его функции (push, pop, isEmpty) с помощью класса односвязного списка и его методов.

Для списка реализованы методы:

- Создание односвязного списка
- Получение адреса самого верхнего элемента
- Ввод данных в верхний элемент списка
- Функция, получающая размер списка
- Добавление нового элемента в список
- Получение адреса определенного элемента по индексу
- Получение индекса по значению внутри элемента
- Получение индекса по значению внутри элемента
- Проверка списка на наличие данных в нем
- Удаление списка и высвобождение памяти

Дополнительно создал класс динамического массива

Для класса динамического массива реализованы методы:

- Обращение к элементу по его индексу
- Запись значения по индексу
- Удаление значения элемента по индексу
- Получение индекса по значению элемента

Алгоритм сортировочной станции был реализован и устроен согласно описанию алгоритма на Википедии. Дополнительно написал проверку на корректность введенных пользователем данных, перед их обработкой функцией сортировочной станции.

## Примеры и демонстрация работы программы:

**Пример 1:**  $3 * (\sin(4) - 1) + 4 * (\cos(2) * 7)$ , с учетом пробелов

```
Type your array: 3 * ( sin ( 4 ) - 1 ) + 4 * ( cos ( 2 ) * 7 )

34sin1-*42cos7**+
C:\3 семестр\First_Lab\x64\Debug\First_Lab.exe (процесс 4988) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

**Пример 2:** проверяем на предупреждение о несовпадении открывающих и закрывающих скобок

(1+3))

```
Type your array: (1+3))
13+
Error. Incorrect data entered

C:\3 семестр\First_Lab\x64\Debug\First_Lab.exe (процесс 20188) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

((2+7)

```
Type your array: ((2+7)
27+
Error. Incorrect data entered

C:\3 семестр\First_Lab\x64\Debug\First_Lab.exe (процесс 11824) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

## **Вывод:**

В ходе выполнения работы я научился реализовывать стек на основе таких структур, как односвязный список и динамический массив, а также, благодаря опыту работы, стал лучше понимать работу всех вышеперечисленных структур данных. Дополнительно научился реализовывать алгоритм сортировочной станции