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

ОТЧЕТ

по лабораторной работе №1

по дисциплине «Алгоритмы и структуры данных»

Тема: Списочные структуры. Алгоритм сортировочной станции.

Студентка гр. 1373 _____ Кубрина Е.

Преподаватель _____ Пелевин М. С.

Санкт-Петербург

2022

Цель работы.

Реализовать следующие структуры: односвязный список, динамический массив и стек.

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.

Односвязный список:

Линейный однонаправленный список — это структура данных, состоящая из элементов одного типа, связанных между собой последовательно посредством указателей. Каждый элемент списка имеет указатель на следующий элемент. Последний элемент списка указывает на NULL.

Иллюстрация списка:

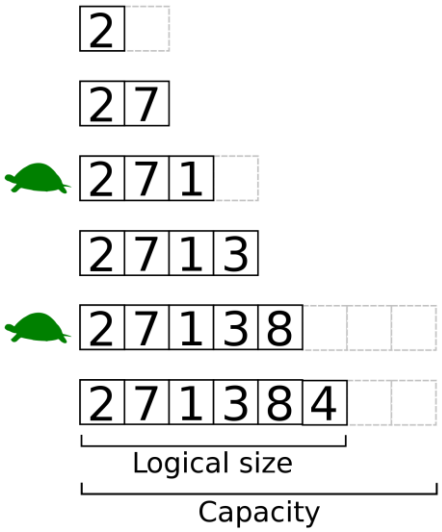
Сложнось работы основных методов списка:

Доступ	Поиск	Вставка	Удаление
O(n)	O(n)	O(1)	O(1)

Динамический массив:

Динамический массив - массив, размер которого может изменяться во время исполнения программы.

Иллюстрация динамического массива:



Сложность основных методов динамического массива:

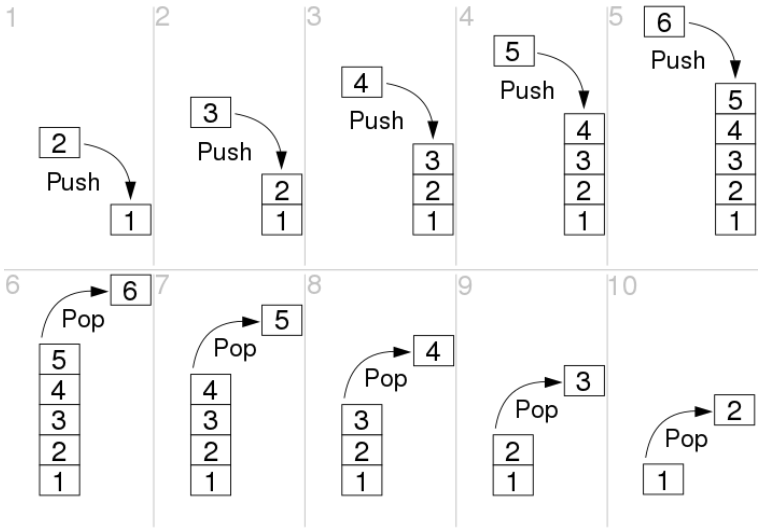
Доступ	O(1)
Поиск	O(n)
Вставка	O(n)
Удаление	O(n)

Стек:

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

Чаще всего принцип работы стека сравнивают со стопкой тарелок: чтобы взять вторую сверху, нужно снять верхнюю.

Иллюстрация работы стека:



Реализация.

Методы динамический массив

- `bool checkForCompletion()`
- `bool checkForCompletion(int _size)`
- `void newMas()`
- `_vector()`
- `_vector(int _size)`
- `int getSize()`
- `int getCapacity()`
- `T& operator [](int index)`
- `int find(T data)`
- `void push_back(T temp)`
- `void push(T temp, int index)`
- `void erase(int index)`
- `~_vector()`

Методы односвязного списка

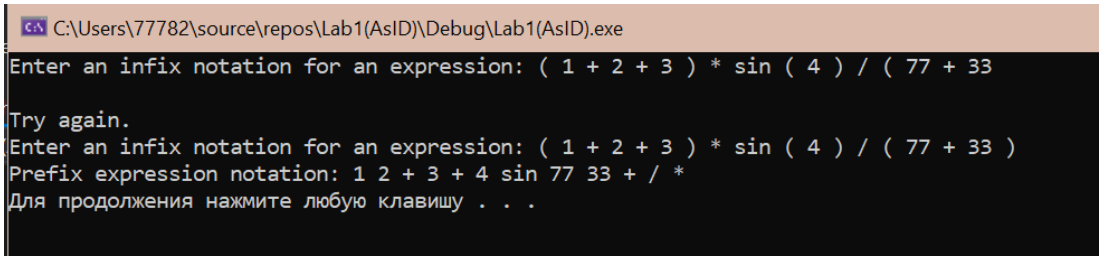
- `_List()`
- `_List(int size)`
- `_List(int size, T data)`
- `_Node* operator[](int index)`

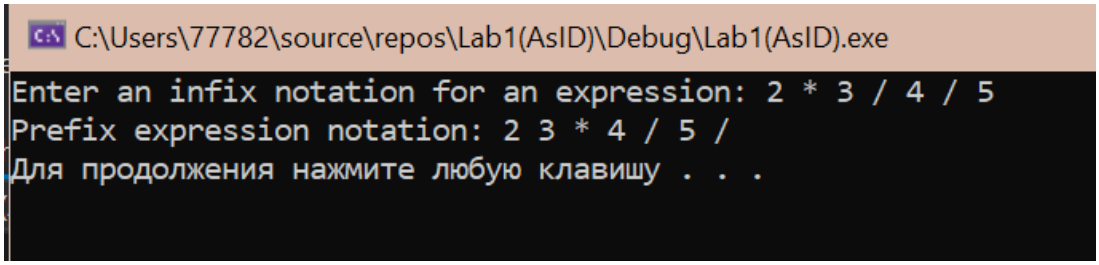
- `T getVal(int index)`
- `void printList()`
- `void printNode(int index)`
- `int find(T data)`
- `int getSize()`
- `bool is_empty()`
- `void push_back(T data)`
- `void push_front(T data)`
- `void push(int index, T data)`
- `void deleteNode(int index)`
- `~_List L pop()`
- `L peak()`
- `void push(L data)`
- `bool is_empty()`

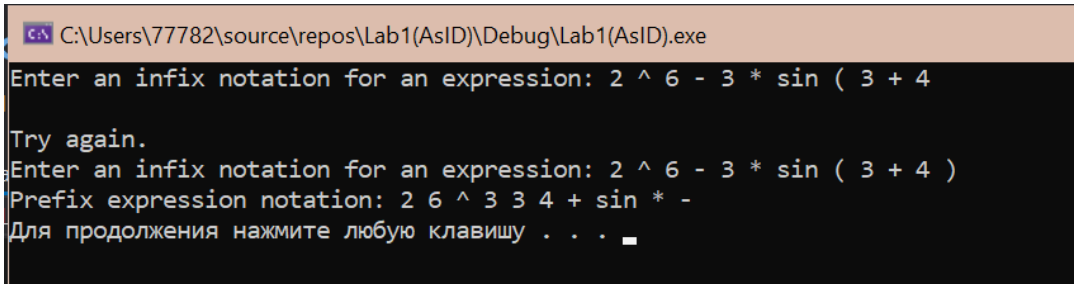
Методы стека:

- `L pop()`
- `void push(L data)`
- `L peak()`
- `bool is_empty()`

Пример работы алгоритма:







Вывод.

Во время проведения этой лабораторной работы я изучила такие структуры данных как односвязный список и динамический массив. Так же я познакомилась с абстрактным типом данных – стеком.

Была произведена реализация односвязного списка, динамического массива, стека и алгоритма сортировочной станции.