Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»

им. В.И. Ульянова (Ленина)»

кафедра систем автоматизированного проектирования

**ОТЧЕТ**

**по лабораторной работе**

**по дисциплине «Программирование»**

**Тема: «Реализация связных списков»**

Выполнил : Носков Евгений Сергеевич

Группа № 1301

Преподаватель: Родионова Е. А.

Санкт-Петербург

2022

Оглавление

[Постановка задачи 3](#_Toc114086034)

[Описание реализуемого класса и методов и оценка временной сложности 3](#_Toc114086035)

[Пример работы программы 7](#_Toc114086036)

[Листинг 7](#_Toc114086037)

# Постановка задачи

Двусвязный список методов/функций, которые реализует каждый вариант (приведено для целочисленного типа данных):

1. добавление в конец списка
2. добавление в начало списка
3. удаление последнего элемента
4. удаление первого элемента
5. добавление элемента по индексу (вставка перед элементом, который был ранее доступен по этому индексу)
6. получение элемента по индексу
7. удаление элемента по индексу
8. получение размера списка
9. удаление всех элементов списка
10. замена элемента по индексу на передаваемый элемент
11. проверка на пустоту списка
12. проверка на содержание другого списка в списке, можно сделать целочисленного типа

# Описание реализуемого класса и методов и оценка временной сложности

Структуры:

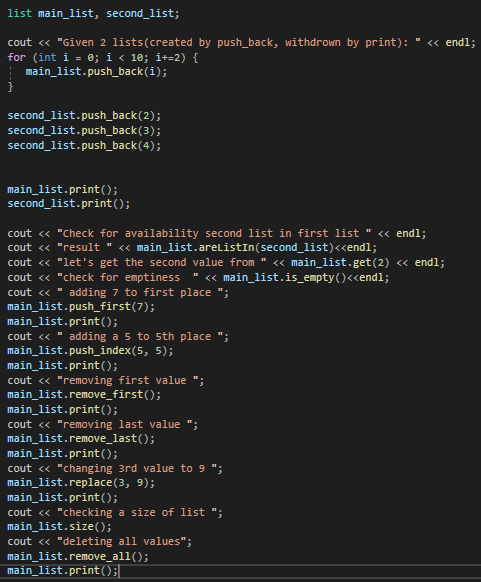
|  |  |
| --- | --- |
| Название и тип | Описание |
| structure Node | Узел списка, в нем хранится данные и ссылки на следующие элементы и 2 метода: переход на следующий элемент и прошлый элемент |
| structure list | Сам список, хранит ссылки на первый и последний элемент. В ней содержатся все нужные методы. |

Методы

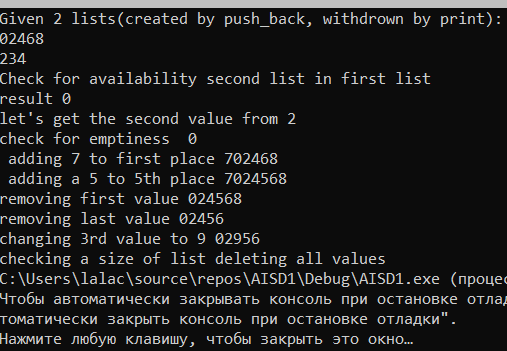
|  |  |  |
| --- | --- | --- |
| Название метода | описание | Оценка временной сложности |
| push\_back | Добавляет в конец списка еще один узел перезаписывая ссылки прошлого последнего элемента, вставляемого элемента и ссылку на последний элемент списка. | O(1) |
| is\_empty() | Проверка на наличие узлов. Проверяется с помощью ссылки на первый элемент. | O(1) |
| push\_first | Добавляет в начало списка еще один узел перезаписывая ссылки следующего первого элемента, вставляемого элемента и ссылку на первый элемент списка. | O(1) |
| push\_index | Смещает узлы перезаписывая ссылки двух соседних узлов, прикрепляя их к новому узлу. В новом узлу так же записываются ссылка на прошлый и следующий элемент. | O(n) |
| remove\_first | Удаляет первый элемент удаляя сам узел с его значением. И меняет ссылку на первый элемент списка и следующий в новом узле. | O(1) |
| remove\_last | Удаляет последний элемент удаляя сам узел с его значением. И меняет ссылку на последний элемент списка и предыдущий в новом узле. | O(1) |
| remove\_index | Удаляет узел, перезаписывая ссылку друг на друга соседних узлов. Удаляет узел. | O(n) |
| remove\_all | Удаляет все узлы, перебирая их с начала до конца и удаляя. | O(n) |
| print | Выводит весь массив, перебирая его и выводя значение в консоль | O(n) |
| get | Выводит значение конкретного узла, доходя до него и выводя его значение. | O(n) |
| replace | Заменяет значение конкретного узла, доходя до него и выводя его значение. | O(n) |
| size | Выводит размер списка, перебирая его и добавляя в счетчик единицу за один переход между узлами. | O(n) |
| areListIn | Проверяет на наличие одного списка в другом( содержаться ли такие же значения в списке, как и в основном). Совершается примитивным пеербором. | O(n^2) |

# Пример работы программы

Был создан набор функции для примера работы программы:



Вывод программы:



# Листинг

https://github.com/Menoitami/AISD-labs/blob/e1dcbcc97a92d3efcbef778ceeb47af0d6687fd5/LAB1/Source.cpp