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 : Герасимов Артемий

Группа № 1301

Преподаватель: Родионова Е. А.

Санкт-Петербург

2022

Оглавление

[Постановка задачи](#_heading=h.1fob9te) **3**

[Описание реализуемого класса и методов и оценка временной сложности](#_heading=h.2et92p0) **3**

[Пример работы программы](#_heading=h.tyjcwt) **5**

[Листинг](#_heading=h.3dy6vkm) **5**

Постановка задачи

Односвязный 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. поиск первого вхождения другого списка в список

Описание реализуемого класса и методов и оценка временной сложности

Структуры:

| Название и тип | Описание |
| --- | --- |
| class Node | Элемент списка, в нем хранятся данные и ссылка на следующий элемент. А также конструктор на вставку элемента |
| class List | Сам список В нем содержатся следующие методы. |

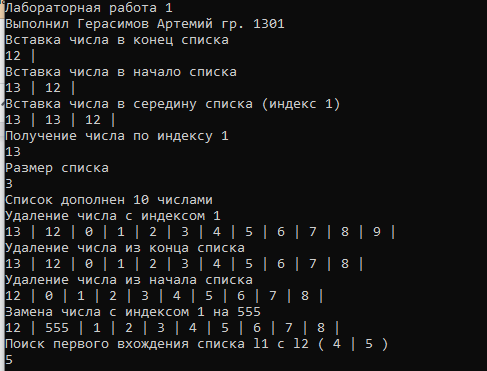
Методы

| Название метода | описание | Оценка временной сложности |
| --- | --- | --- |
| push\_back | Добавляет в конец списка еще один элемент перезаписывая ссылку прошлого последнего элемента. | O(1) |
| check | Проверка на наличие узлов. | O(1) |
| push\_front | Добавляет в начало списка еще один узел, вставляя в него ссылку на прошлый первый элемент. | O(1) |
| del\_front | Удаляет первый элемент удаляя сам узел с его значением. | O(1) |
| del\_back | Удаляет последний элемент. | O(1) |
| del\_mid | Удаляет элемент под выбранным номером. При этом ссылка предыдущего элемента перезаписывается на следующий | O(n) |
| replace | Заменяет выбранный узел на новый, со ссылкой на следующий, перезаписывая ссылку прошлого. | O(n) |
| clear | Удаляет все узлы, перебирая их с начала до конца и удаляя. | O(n) |
| get\_index | Получает узел с заданным индексом | O(n) |
| push\_mid | Вставляет новый узел на выбранное место, меняя ссылку текущего элемента на новый | O(n) |
| get\_size | Возвращает размер списка. | O(n) |
| compare | Ищет первое вхождение одного списка в другой, путем перебора всех узлов | O(n) |

Пример работы программы

В файле [Source.cpp](https://github.com/ArtGer2/Frogram/blob/main/Source.cpp) написан ряд проверок на каждую из написанных функций.

Вывод программы:



Листинг

https://github.com/ArtGer2/Frogram/blob/main/Source.cpp