**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

Кафедра систем автоматизированного проектирования

Отчет по лабораторной работе № 1(вариант 8)

по дисциплине «Алгоритмы и структуры данных»

|  |  |
| --- | --- |
| Студент гр. 9301 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Игнатьев.К.А |
|  |  |
| Преподаватель | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Тутуева А.В. |

**Оглавление**

[Постановка задачи 2](#_Toc50933615)

[Описание реализуемого класса и методов 2-4](#_Toc50933616)

[Описание реализованных unit-тестов 4](#_Toc50933617)-5

[Пример работы программы 6](#_Toc50933617)

[Текст программы 7](#_Toc50933617)

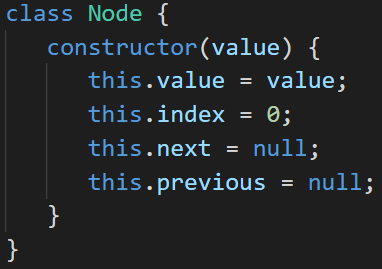
[Заключение 8](#_Toc50933618)

**Постановка задачи:**

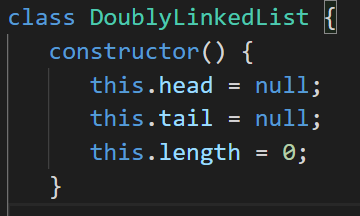
Реализовать класс связного списка с набором методов. Данные, хранящиеся в списке могут быть любого типа. Язык программирования – javascript.

**Описание реализуемого класса и методов:**

Описание класса узла списка:



Описание класса списка:



+ далее идет описание методов

Описание методов списка:

|  |  |  |
| --- | --- | --- |
| Название метода | Что делает | Временная сложность |
| get\_size() | Возвращает размер списка | O(1) |
| isEmpty() | Возвращает true, если список пуст, и false, если в списке есть хотя бы один узел | O(1) |
| push\_back(value) | Добавляет узел в конец списка | O(1) |
| at(initial\_index) | Возвращает узел по индексу | O(n) |
| increases\_indexes(initial\_index) | Увеличивает все индексы начиная с определенного(initial\_index) на единицу | O(n) |
| reduces\_indexes(index) | Уменьшает все индексы начиная с определенного(index) на единицу | O(n) |
| push\_front(value) | Добавляет узел в начало списка | O(n)(так как необходимо менять индексы) |
| print\_to\_console(separator) | Выводит список в консоль через заданный пользователем разделитель(separator) | O(n) |
| pop\_back() | Удаляет последний узел списка | O(1) |
| pop\_front() | Удаляет первый узел списка | O(n) |
| insert(value, index) | Вставляет узел перед узлом с заданным индексом(index) | insert() - O(n) или O(1)(если не надо менять индексы) |
| remove(index) | Удаляет элемент по индексу | O(n) или O(1)(если не надо менять индексы) |
| set(value, index) | Заменяет элемент по индексу на передаваемый элемент | O(n) |
| clear() | Удаляет все элементы списка | O(n) |
| push\_front\_list(list) | Вставляет один список в начало другого | O(n)(так как необходимо менять индексы) |

**Описание реализованных unit-тестов:**

isEmpty():

Тестируем как метод работает с пустым списком, с непустым списком

get\_size():

Тестируем как метод работает с пустым списком, с непустым списком

push\_back():

Тестируем добавление элемента в пустой список, в непустой список

at():

Проверяем, соответствуют ли возвращаемые методом индексы действительности

push\_front():

Тестируем добавление элемента в пустой список, в непустой список

pop\_back():

Тестируем удаление из списка, где > 1 элемента, где 1 элемент

pop\_front():

Тестируем удаление из списка, где > 1 элемента, где 1 элемент

insert():

Проверяем, что элемента был вставлен в нужное место

remove():

Проверяем, что элемент действительно удален

set():

Проверяем, заменил ли передаваемый элемент ранее доступный по этому же индексу элемент

clear():

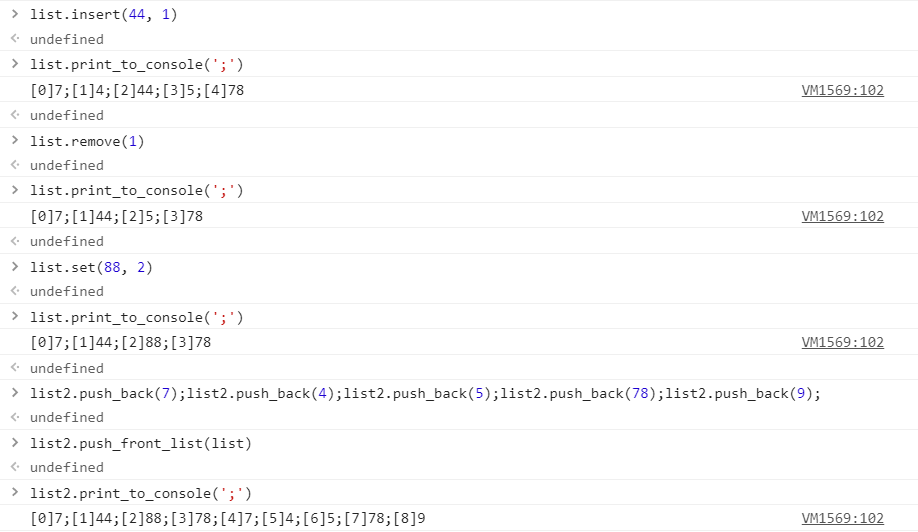
Проверяем, что удалены все элементы списка

push\_front\_list():

Проверяем, был ли вставлен один список в начало другого

**Пример работы программы**

В консоли браузера Google Chrome:

**** ****

**Текст программы:**

Ссылка на гитхаб: https://github.com/kostyan87/lab1\_ignatev\_9301

**Заключение**

В процессе лабораторной работы были получены практические навыки работы со структурой данных «Двусвязный список».