Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра компьютерных систем в управлении и проектировании (КСУП)

**Двусвязный список**

Лабораторная работа №2 по

Дисциплине «Структуры данных»

|  |  |
| --- | --- |
|  | Студент гр. 571-2  \_\_\_\_\_\_\_\_\_\_\_ К.Д. Никитин  (подпись)  \_\_\_\_\_\_\_\_\_\_\_  (дата)  Руководитель  к.т.н., доцент каф. КСУП |
| \_\_\_\_\_\_\_\_\_\_\_  (дата)  М.П. | \_\_\_\_\_\_\_\_\_\_\_ А.А. Калентьев  (подпись)  \_\_\_\_\_\_\_\_\_\_\_  (дата) |

## Содержание

1 Введение………………………………………………………………………..3

2 Основная часть………………………………………………………………....4

2.1 Описание функций…………………………………………………….4

2.2 Исследование сложности алгоритмов………………………………..5

3 Заключение……………………………………………………………………..9

## 1 ВВЕДЕНИЕ

Цель работы: реализовать структуру данных «Двусвязный список» и набор функций для работы с ней.  
Необходимо обеспечить безопасность функций и всей программы в целом.  
Необходимо реализовать следующие функции:

• Функция создания и инициализации полей списка;  
• Добавления элемента;  
• Удаление элемента;  
• Вставка элемента в начало;  
• Вставка элемента в конец;  
• Вставка после определенного элемента;  
• Вставка перед определенным элементом;  
• Сортировка списка;  
• Линейный поиск элемента в двусвязном списке.

## 2 ОСНОВНАЯ ЧАСТЬ

## 2.1 Описание функций

AddElement – добавляет элемент в конец списка и в структуре двусвязного списка устанавливает указатель этого элемента в качестве последнего. Сложность О(1).

RemoveElement – удаляет элемент по переданном индексу. При удалении начального элемента указатель на голову списка перемещается на следующий элемент. При удалении последнего элемента, указатель на последний элемент перемещается на предыдущий.

InsertElementInBegin – вставка элемента в начало. Указатель начала списка заносится адрес этого элемента. Сложность О(1).

InsertElementBeforeCertainElement – вставка элемента перед переданным в метод индексом. Сложность О(n).

InsertElementAfterCertainElement – вставка элемента после переданного в метод индекса. Сложность О(n).

SortList – сортировка списка. Сложность О(n2 ).

LinearSearch – производит поиск элемента по указанному индексу. Возвращает число больше или равно 0, если это индекс искомого числа, иначе возвращает -1, указывая, что элемента в списке нет.

## 2.2 Исследование сложности алгоритмов

# 1. Оценка сложности алгоритма для операции вставки элемента перед указанным индексом.

На рисунке 2.1 изображен график, показывающий отношение времени выполнения алгоритма вставки элемента в середину списка к количеству элементов в двусвязном списке.

(ИСПРАВИТЬ ПОДПИСЬ на рис)



Рис. 2.1. Отношение между временем выполнения алгоритма и кол-вом элементов.

Таблица 2.1. Отношение между временем выполнения алгоритма и кол-вом элементов.

|  |  |
| --- | --- |
| Время, мкс. | Кол-во элементов, ед. |
| 2145 | 100000 |
| 6957 | 250000 |
| 16726 | 500000 |
| 22816 | 750000 |
| 31836 | 1000000 |

По графику видно, что сложность данного алгоритма равна О(n).

### Оценка сложности алгоритма для операции вставки элемента в начало.

На рисунке 2.2 изображен график, показывающий отношение времени выполнения алгоритма вставки элемента в начало списка к количеству элементов в этом двусвязном списке. (ПОДПИСЬ НА РИСУНКЕ)



Рис. 2.2. Отношение между временем выполнения алгоритма и кол-вом элементов.

Таблица 2.2. Результат работы алгоритма вставки элемента в начало.

|  |  |
| --- | --- |
| Время, мкс. | Кол-во элементов, ед. |
| 1 | 100000 |
| 1 | 250000 |
| 1 | 500000 |
| 1 | 750000 |
| 1 | 1000000 |

По графику видно, что сложность данного алгоритма равна О(1).

### Операция удаление элемента из списка.

На рисунке 2.3 изображен график, показывающий отношение времени выполнения алгоритма удаления элемента к количеству элементов в двусвязном списке. (ПОДПИСЬ НА РИС)

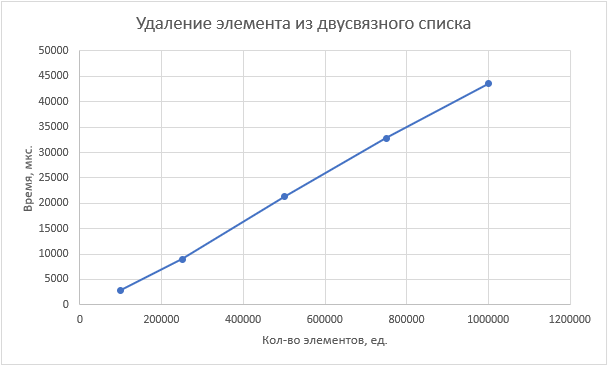


Рис 2.3. Отношение между временем выполнения алгоритма и кол-вом элементов.

Таблица 2.3. Результаты работы алгоритма удаления элемента.

|  |  |
| --- | --- |
| Время, мкс. | Кол-во элементов, ед. |
| 2836 | 100000 |
| 8957 | 250000 |
| 21306 | 500000 |
| 32850 | 750000 |
| 43539 | 1000000 |

По графику видно, что сложность данного алгоритма равна О(n).

# 3 Заключение

В ходе данной лабораторной работы, мной был разработана структура данных двусвязный список, а также базовые функции для работы с ним.

Были проведены исследования, в ходе которых я установил сложность некоторых алгоритмов:

1. Алгоритм вставки элемента перед определенным индексом имеет сложность O(n).
2. Алгоритм вставки элемента в начало имеет сложность O(1).
3. Алгоритм удаления элемента имеет сложность O(n).