



МИНОБРНАУКИ РОССИИ

*Федеральное государственное бюджетное образовательное учреждение высшего  
образования*

*«МИРЭА – Российский технологический университет»*

**РТУ МИРЭА**

---

Отчет по выполнению практического задания № 5

**Тема:**

«Однонаправленный динамический список»

Дисциплина: «Структуры и алгоритмы обработки данных»

Выполнил студент: Боргачев Т.М.

Группа: ИНБО-10-23

Москва – 2024

## СОДЕРЖАНИЕ

1 ФОРМУЛИРОВКА ЗАДАЧИ .....	3
1.1    Задание.....	3
2  ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ .....	4
2.1  Определение операций над списком.....	4
2.1.1  Определение структуры узла однонаправленного списка.....	4
2.1.2  Изображение полученного списка .....	4
2.1.3  Изображение структуры данных .....	5
2.1.4  Алгоритм выполнения операции.....	6
2.1.5  Тесты алгоритмов.....	6
2.2  Код программы .....	6
2.3  Результат тестирования программы .....	6
3  ВЫВОДЫ .....	8
4  ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ.....	8

Цель: получить знания и практические навыки управления динамическим однонаправленным списком.

## **1 ФОРМУЛИРОВКА ЗАДАЧИ**

### **1.1 Задание**

Реализуйте программу решения задачи варианта по использованию линейного однонаправленного списка.

Требования для всех вариантов:

1. Информационная часть узла определена вариантом;
2. Разработать функции вставки нового узла перед первым узлом и удаления узла по ключу;
3. Реализовать возможность а) создания нового списка вручную, а также б) использования уже готового списка для тестирования заданий индивидуального варианта;
4. Разработать функцию вывода списка в консоль;
5. Разработать функции согласно индивидуальному варианту. При необходимости можно добавлять вспомогательные функции, декомпозируя задачу;
6. Реализовать текстовое пользовательское меню;
7. В основной программе выполните тестирование каждой функции.

Требования для индивидуального варианта 5:

Даны два линейных однонаправленных списка L1 и L2 с головным элементом.

1. Разработать функцию, которая проверяет на равенство списки L1 и L2;
2. Разработать функцию, которая вставляет в список L1 последний элемент списка L2;
3. Разработать функцию, которая удаляет из списка L2, узлы, содержащие цифровые значения.

## 2 ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ

### 2.1 Определение операций над списком

#### 2.1.1 Определение структуры узла однонаправленного списка

В соответствии с заданием, список будет содержать произвольные символы типа `char`, а также указатели на следующий узел. Для ввода данных в списки целесообразно использовать `txt` файлы, чтобы их можно было в любой момент поменять.

#### 2.1.2 Изображение полученного списка

Блок-схемы функций проверки списков на равенство, удаления из списка цифр и вставки последнего элемента одного массива в другой представлены на рис. 1, 2 и 3 соответственно.

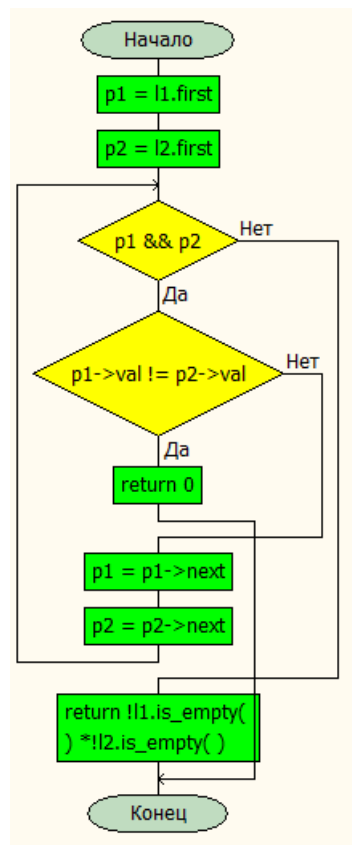


Рисунок 1 – Функция сравнения списков

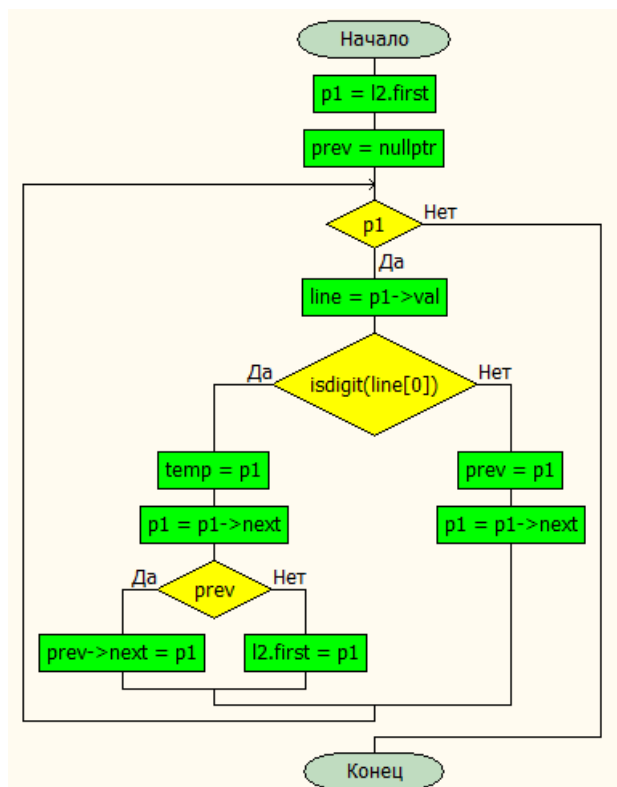


Рисунок 2 – Функция удаления из списка цифр

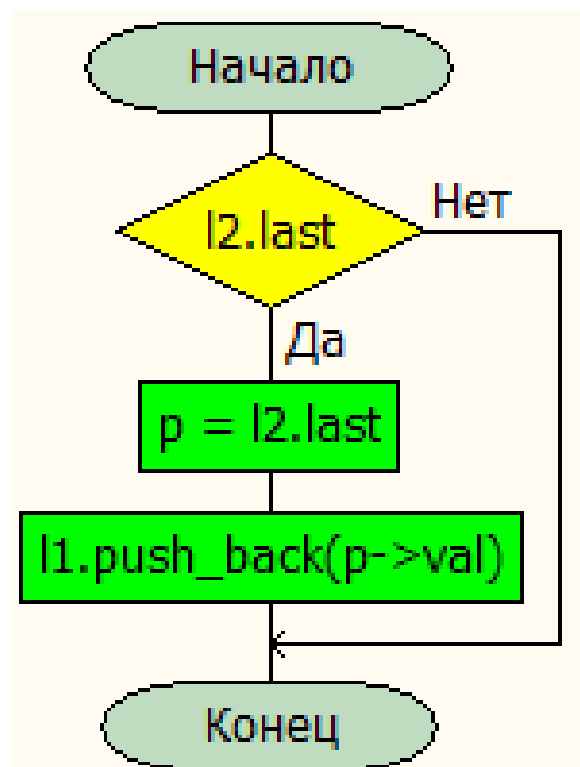


Рисунок 3 – Функция вставки последнего элемента

### 2.1.3 Изображение структуры данных

Схематическое изображение односвязного списка представлено на рис. 4.



Рисунок 4 – Структура данных списка

#### 2.1.4 Алгоритм выполнения операции

Для реализации и проверки функций, создадим два списка с символьными и числовыми значениями, проверяем списки на равенство, создаем список копию первого – проверяем их на равенство, вызываем метод вставки для первого списка, а метод удаления цифр для второго списка.

#### 2.1.5 Тесты алгоритмов

Ожидаемые результаты работы алгоритмов представлены в табл. 1.

Таблица 1 – Ожидаемы результаты

Списки	Первый f d s e 5 b	Второй l l d s t 2	Третий f d s e 5 b
Равенство со след	Нет (2)	Нет (3)	Да (1)
Вставка	f d s e 5 b 2	l l d s t 2	f d s e 5 b
Удаление чисел	f d s e b	l d s t	f d s e b

## 2.2 Код программы

Реализуем функции на языке программирования C++. Функции представлены на рис 5, 6 и 7.

```
bool ravenstvo(list &l1, list &l2) {
    Node* p1 = l1.first;
    Node* p2 = l2.first;
    while (p1 && p2) {
        if (p1->val != p2->val) {
            return 0;
        }
        p1 = p1->next;
        p2 = p2->next;
    }
    return !l1.is_empty()*!l2.is_empty();
}
```

Рисунок 5 – Функция проверки на равенство

```
void insert_last (list &l1, list &l2) {
    if (l2.last) {
        Node* p = l2.last;
        l1.push_back(p->val);
    }
}
```

Рисунок 6 – Функция вставки

```

void delete_chisla(list &l2) {
    Node* p1 = l2.first;
    Node* prev = nullptr;
    while (p1) {
        string line = p1->val;
        if (isdigit(line[0])) {
            Node* temp = p1;
            p1 = p1->next;
            delete temp;
            if (prev)
                prev->next = p1;
            else
                l2.first = p1;
        }
        else {
            prev = p1;
            p1 = p1->next;
        }
    }
}

```

Рисунок 7 – Функция удаления чисел из списка

## 2.3 Результат тестирования программы

Результаты тестирования программы представлены на рис. 8.

```

List 1:
f d s e 5 b j f 4 8 w 1 c m z 1 y 8 k o 8 6 r 4 3 1 s x v n m
List 2:
l d s t b v c g h u y t 8 1 2 s 0 h z 4 n 6 d n m k 7 h t 4 2
Списки не равны
List 3:
f d s e 5 b j f 4 8 w 1 c m z 1 y 8 k o 8 6 r 4 3 1 s x v n m
Списки равны
Список 1 после вставки:
f d s e 5 b j f 4 8 w 1 c m z 1 y 8 k o 8 6 r 4 3 1 s x v n m 2
Список 2 после удаления чисел
l d s t b v c g h u y t s h z n d n m k h t

```

Рисунок 8 – Результаты тестирования программы

Исходя из результатов тестирования, можно сделать вывод о корректной работе алгоритмов.

### 3 ВЫВОДЫ

В ходе работы были реализованы структуры для работы с однонаправленными списками, написаны методы для работы с данными в них, а также 3 функции для обработки данных, а именно: сравнение на равенство, удаление цифр и вставка последнего элемента.

Таким образом, были получены теоретические знания о двунаправленных списках, а также практические навыки и умения для работы с ними, и их реализации в программах на C++.

### 4 ИНФОРМАЦИОННЫЕ ИСТОЧНИКИ

1. Сартаков М.В., ПР-1.1 (Теоретическая сложность алгоритма)М., МИРЭА — Российский технологический университет – 12 с. - URL: [https://online-edu.mirea.ru/pluginfile.php?file=%2F1042738%2Fmod\\_assign%2Fintroattachment%2F0%2FПР1.1%20%28Теоретическая%20сложность%20алгоритма%29.pdf&amp;forcedownload=1](https://online-edu.mirea.ru/pluginfile.php?file=%2F1042738%2Fmod_assign%2Fintroattachment%2F0%2FПР1.1%20%28Теоретическая%20сложность%20алгоритма%29.pdf&amp;forcedownload=1) (дата обращения: 15.02.2024). - Режим доступа: Электронно-облачная система – Cloud MIREA РТУ МИРЭА. - Текст: электронный.

2. Рысин М.Л., Сартаков М.В., Туманова М.Б., Введение в структуры и алгоритмы обработки данных. Ч. 1 - учебное пособие, 2022, МИРЭА – Российский технологический университет. – 2022, 109с. – URL: <file:///C:/Users/borga/Downloads/Рысин%20М.Л.%20и%20др.%20Введение%20в%20структуры%20и%20алгоритмы%20обработки%20данных.%20Ч.%201%20-%20учебное%20пособие,%202022.pdf> (дата обращения: 15.02.2024 ). – Режим доступа: Электронно-облачная система – Cloud MIREA РТУ МИРЭА. - Текст: электронный.