|  |
| --- |
|  |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования  **«МИРЭА** - **Российский технологический университет»**  **РТУ МИРЭА** |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №2**  Стандартные типы данных для представления многоэлементных не однородных структур данных | |
| **по дисциплине** | |
| «Структуры и алгоритмы обработки данных» | |
| Выполнил студент группы ИКБО-02-22 | Вагулич А.Р. |
|  |  |
| Принял | Сорокин А.В. |
|  |  |

Практическая работа выполнена «\_\_» 2023 г.

(подпись студента)

«Зачтено» «\_\_» 2023 г.

(подпись преподавателя)

Москва 2023

**1 Постановка задачи**

Вариант 2

Даны два линейных однонаправленных списка L1 и L2. Разработать функцию для создания исходного списка, используя функцию вставки нового узла перед первым узлом. Разработать функцию вывода списка. Разработать функцию, которая формирует список L, включив в него по одному разу элементы, значения которых входят одновременно в оба списка L1 и L2. Разработать функцию, которая удаляет узел списка L2, расположенный перед узлом, содержащим отрицательное значение. И так для всех узлов, содержащих отрицательное значение. Разработать функцию, которая вставляет новый узел с заданным значением перед каждым узлом списка L1, содержащим нечетное значение.

**2 Модель решения**

Список операций над списками:

1. Создание списков L1 и L2
2. Поиск одинаковых элементов списков L1 и L2
3. Внесение в список L одинаковых элементов списков L1 и L2
4. Удаление узла списка L2
5. Вставка нового узла в список L1

**2.1 Определить структуру узла однонаправленного списка в соответствии с вариантом.**

struct Node {

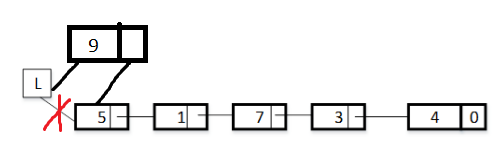
float data;

Node\* next;

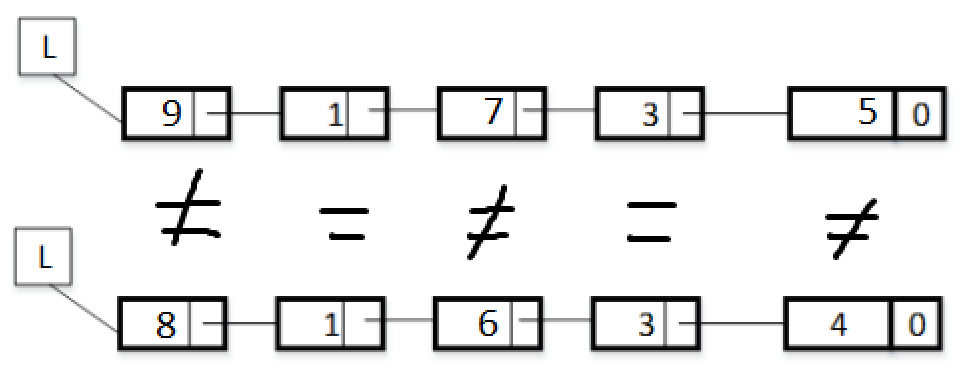
};

**2.2 Изобразить (рисунок) для каждой операции полученного списка процесс выполнения операции на существующем однонаправленном списке.**

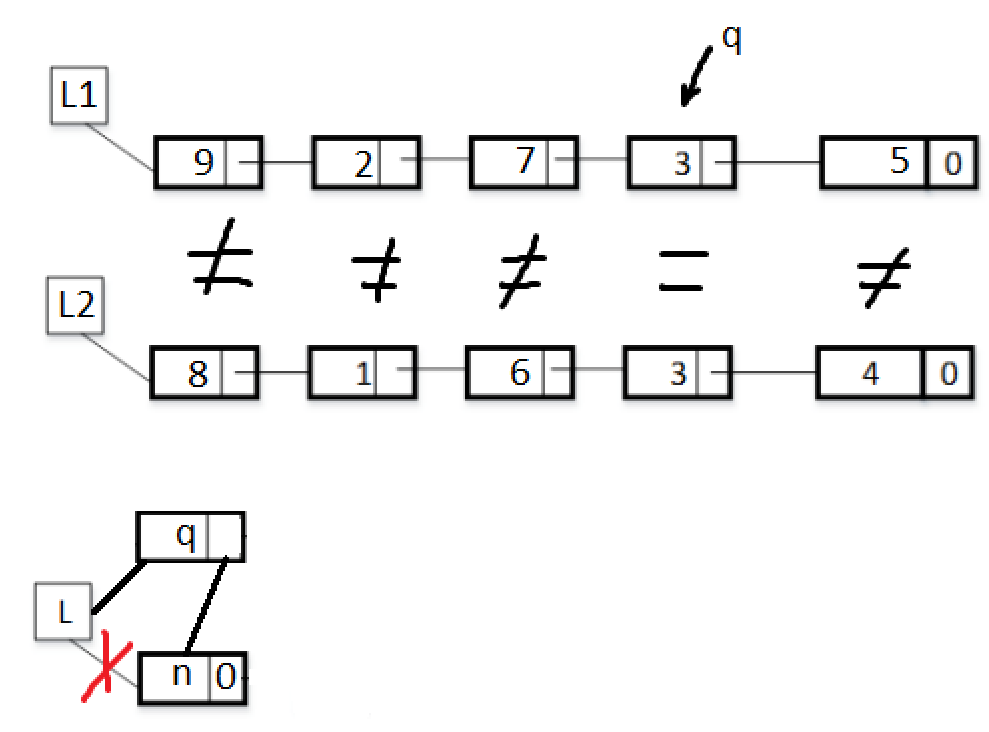
1) Создание списка L1 (или L2)



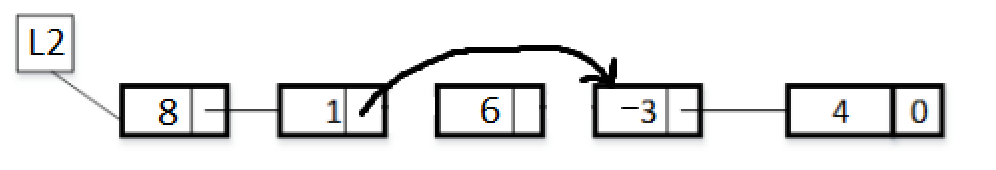
2) Поиск одинаковых элементов списков L1 и L2



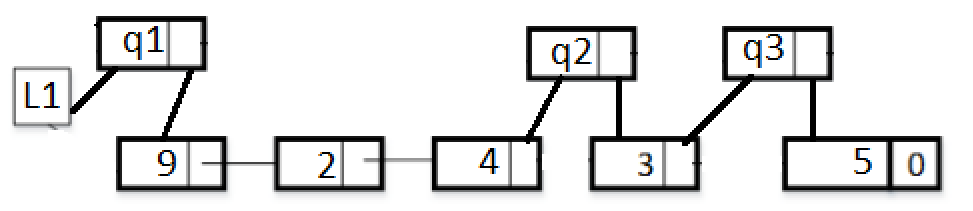
3) Внесение в список L одинаковых элементов списков L1 и L2



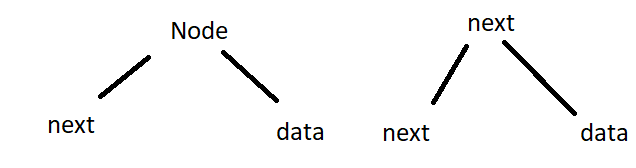
4) Удаление узла списка L2 расположенный перед узлом, содержащим отрицательное значение



5) Вставка узла в список L1 перед каждым узлом списка, содержащим нечетное значение



**2.3. Изобразить структуру данных, которая будет использоваться в операциях.**

****

**2.4. Пример алгоритма выполнения операции**

1) Создание списков, добавляя последующие элементы перед первым

2) Создание списка, добавляя общие элементы первых двух списков

3) Удаление узла в списке L2, расположенного перед узлом, содержащим отрицательное значение; Создаем связь элемента до удаляемого и после удаляемого

4) Добавление узла в список L1 перед узлом, содержащим нечетное значение

**2.5 Таблица тестов для тестирования каждой операции**

1) Создание списков

1.1) Создание списка L1

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| 8.15 | -94.125 13.09 -1.24 3.17 -5.74 16.08 -3.32 8.15 |
| -3.32 |  |
| 16.08 |  |
| -5.74 |  |
| 3.17 |  |
| -1.24 |  |
| 13.09 |  |
| -94.125 |  |

1.2) Создание списка L2

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| 13.09 | -0.56 50.50 -12.12 5.7 -10.01 3.17 -5.74 13.09 |
| -5.74 |  |
| 3.17 |  |
| -10.01 |  |
| 5.7 |  |
| -12.12 |  |
| 50.50 |  |
| -0.56 |  |

2) Создание списка L

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
|  |  |

3) Удаление узла в списке L2

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
|  |  |

4) Добавление узла в список L1

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
|  |  |