Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Кафедра **«**Информационные технологии и автоматизированные системы**»**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №11**

Дисциплина: «Основы алгоритмизации и программирования»

Семестр 2

Тема: Информационные динамические структуры.

Вариант 25.

Выполнил работу

Студент группы РИС-22-1Б

Поважный В. Е.

Проверил

Доцент кафедры ИТАС

Полякова О.А.

г. Пермь-2023

**Введение**

Для выполнения лабораторной работы требуется сделать отчет программы. Создание программ – отличный способ практики программирования.

**Постановка задачи**

Записи в линейном списке содержат ключевое поле типа \*char (строка символов). Сформировать двунаправленный список. Удалить элемент с заданным ключом. Добавить К элементов перед элементом с заданным номером.

**Код программы**

#include <iostream>

#include <cstring>

class List

{

public:

List() : root(0) {}

void Add(char const\* rhs)

{

if (!root)

root = new Node(0, rhs);

else

root->Add(rhs);

}

void Erase(char const\* rhs)

{

if (root)

if (!strcmp(root->pstr, rhs))

{

Node\* tmp = root;

root = root->next;

if (root)

root->prev = 0;

tmp->prev = 0;

tmp->next = 0;

delete tmp;

}

else

root->Erase(rhs);

}

~List()

{

delete root;

root = 0;

}

private:

struct Node

{

Node(Node\* pnode) : prev(pnode), next(0), pstr(0) {}

Node(Node\* pnode, char const\* rhs) : prev(pnode), next(0)

{

pstr = new char[strlen(rhs) + 1];

strcpy(pstr, rhs);

}

void Add(char const\* rhs)

{

if (!next)

next = new Node(this, rhs);

else

next->Add(rhs);

}

void Erase(char const\* rhs)

{

if (!strcmp(rhs, pstr))

{

std::cout << "Found\n";

prev->next = next;

if (next)

next->prev = prev;

prev = 0;

next = 0;

delete this;

std::cout << "Erased\n";

}

else if (next)

next->Erase(rhs);

}

~Node()

{

if (next)

delete next;

prev = 0;

next = 0;

std::cout << "Destructor: " << pstr << '\n';

delete[] pstr;

pstr = 0;

}

Node\* prev, \* next;

char\* pstr;

private:

Node(Node const&);

Node& operator=(Node);

};

Node\* root;

List(List const&);

List& operator=(List);

friend struct Node;

};

int main()

{

List list;

int n;

std::cin >> n;

for (int i = 0; i < n; i++)

{

std::cout << i + 1 << ". ";

std::string line;

std::cin >> line;

list.Add(line.c\_str());

}

std::string s;

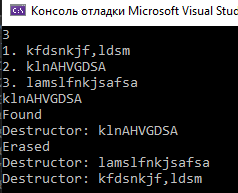
std::cin >> s;

list.Erase(s.c\_str());

return 0;

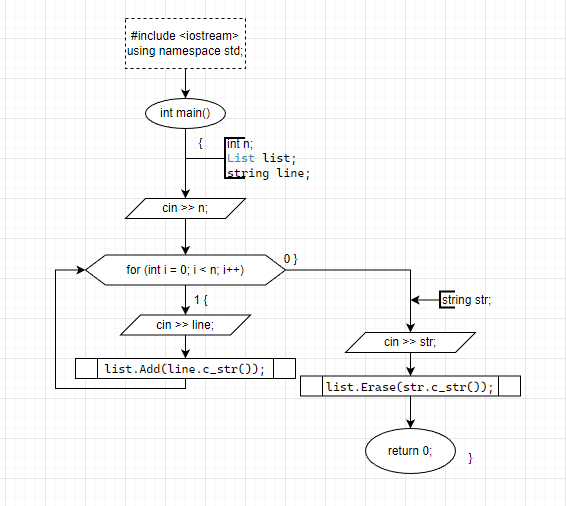
}

**Вывод программы**

****

***Рисунок 1 – вывод программы***

**Блок-схема**

****

***Рисунок 3 – Схема алгоритма программы***

**Вывод**

Программа выполняет свою задачу.

**Заключение**

Для решения задачи потребовались знания языка программирования, а конкретнее C++. Программа выполняет те условия, что были указаны в постановке задачи и работает без проблем.