Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Е Т  
по лабораторной работе №11.2**

**по дисциплине «Основы алгоритмизации и программирования»**

Выполнил студент гр. РИС-21-1б

Ибрагимов Марк Магамедович

(Фамилия, Имя, Отчество)

(подпись)

Проверил:

доцент

кафедры ИТАС

Полякова О.А.

(оценка) (подпись)

(дата)

Пермь 2022

**Постановка задачи**

Тип информационного поля int. Удалить из списка все элементы с четными информационными полями.

**Анализ задачи**

1. Необходимо создать список из элементов типа int. Каждый элемент (узел) содержит информационное поле и поле с ссылкой на следующий элемент (или nullptr) или на предыдущий (или nullptr).
2. Необходимо добавить несколько узлов списка, поочерёдно перенаправлять ссылки этих элементов.
3. Через цикл for необходимо найти элемент, остаток от деления на 2 которого будет равен 0. Такой узел необходимо удалить из списка, путём переадресации ссылок.
4. Полученный массив необходимо вывести на экран.

**Описание переменных**

list list – список состоящий из переменных структуры list.

point\* el/elem – переменный нового узла списка.

int data – данные в узле (информационное поле).

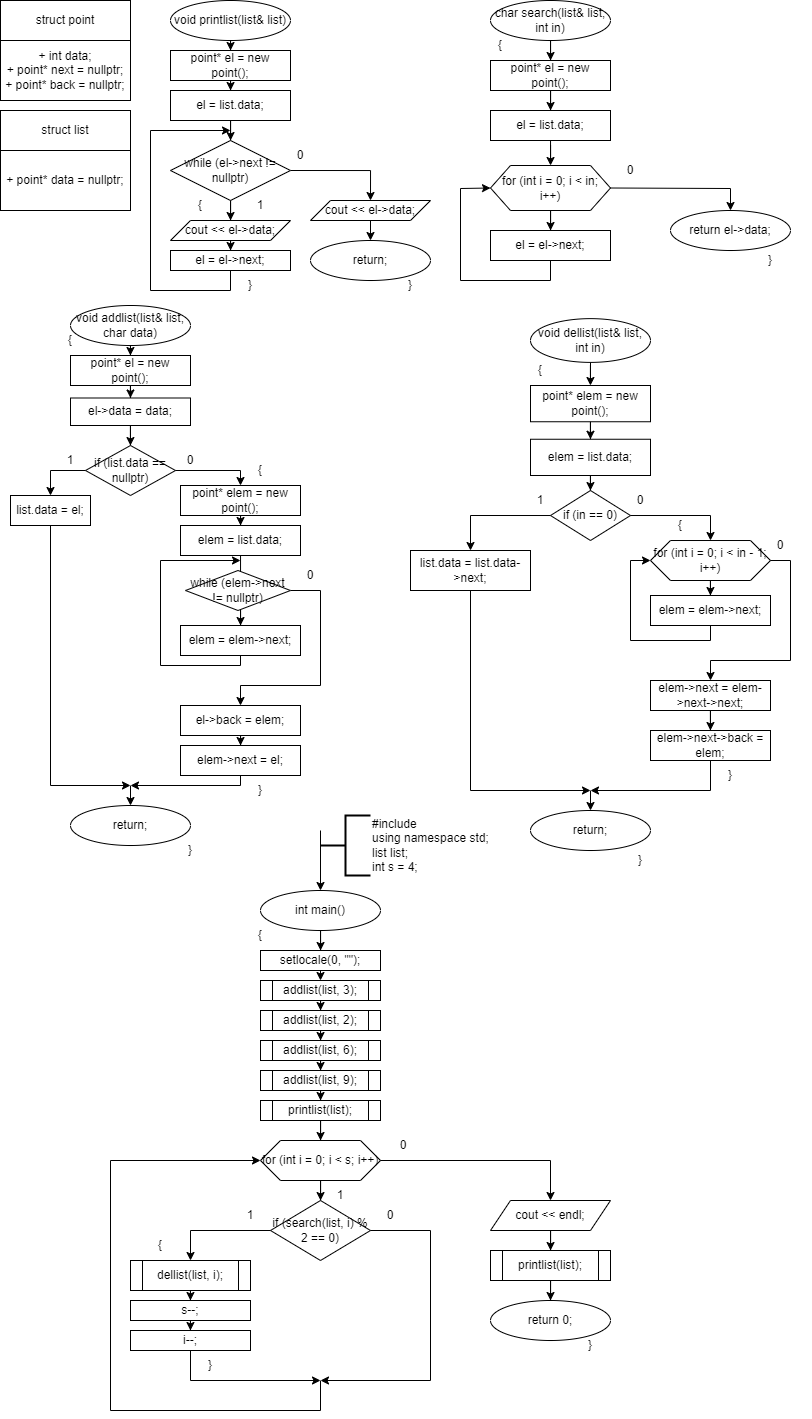
point\* next – ссылка на следующий узел списка;

point\* back – ссылка на предыдущий узел списка;

int i – переменная, используемая как счётчик в циклах.

int s – количество узлов в списке.

**Блок схема**

****

**Код программы**

#include <iostream>

using namespace std;

struct point

{

int data;

point\* next = nullptr;

point\* back = nullptr;

};

struct list

{

point\* data = nullptr;

};

char search(list& list, int in)

{

point\* el = new point();

el = list.data;

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

{

el = el->next;

}

return el->data;

}

void addlist(list& list)

{

point\* el = new point();

int data;

cout << "Введите элемент списка: ";

cin >> data;

el->data = data;

if (list.data == nullptr)

{

list.data = el;

}

else

{

point\* elem = new point();

elem = list.data;

while (elem->next != nullptr)

{

elem = elem->next;

}

el->back = elem;

elem->next = el;

}

}

void printlist(list& list)

{

point\* el = new point();

cout << "Получившийся список: ";

el = list.data;

while (el->next != nullptr)

{

cout << el->data << ' ';

el = el->next;

}

cout << el->data << ' ';

}

void dellist(list& list, int in)

{

point\* elem = new point();

elem = list.data;

if (in == 0) {

list.data = list.data->next;

}

else {

for (int i = 0; i < in - 1; i++)

{

elem = elem->next;

}

elem->next = elem->next->next;

elem->next->back = elem;

}

}

int main()

{

setlocale(0, "");

list list;

int n;

cout << "Введите количество элементов списка: ";

cin >> n;

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

{

addlist(list);

}

printlist(list);

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

{

if (search(list, i) % 2 == 0)

{

dellist(list, i);

n--;

i--;

}

}

cout << endl;

printlist(list);

return 0;

}

**Примеры выполнения программы**

**Изображение выглядит как текст

Автоматически созданное описание**

**Анализ результатов**

Программа сработала корректно и вывела необходимые результаты.