Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЁТ**

**по лабораторной работе №11.2**

Дисциплина: «Основы алгоритмизации и программирования»

Тема: Информационные и динамические структуры. Двунаправленные списки

Вариант 21

Выполнил:

Студент группы ИВТ-20-1б

Углицких Максим Сергеевич

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь 2021**

**Цель работы:** получить практические навыки работы с двунаправленными списками.

**Постановка задачи:**

* Сформировать двунаправленный список.
* Распечатать полученную структуру.
* Выполнить обработку структуры в соответствии с заданием.
* Распечатать полученный результат.
* Удалить соответствующую структуру из памяти.

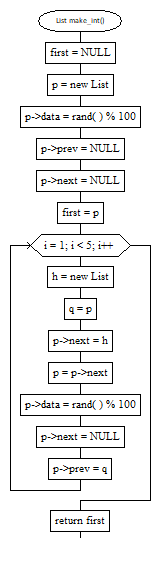
**Вариант:** 21

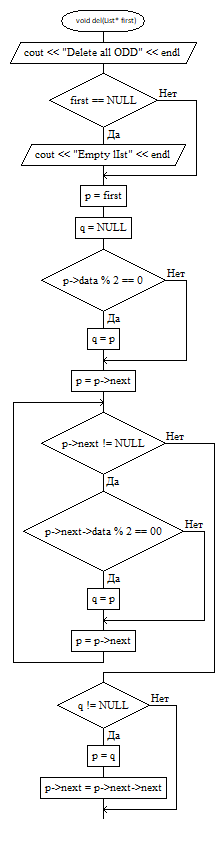
Тип информационного поля int. Удалить из списка все элементы с четными информационными полями.

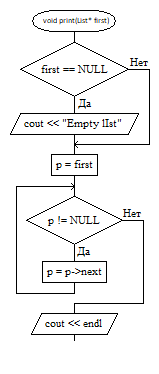
**Анализ задачи:**

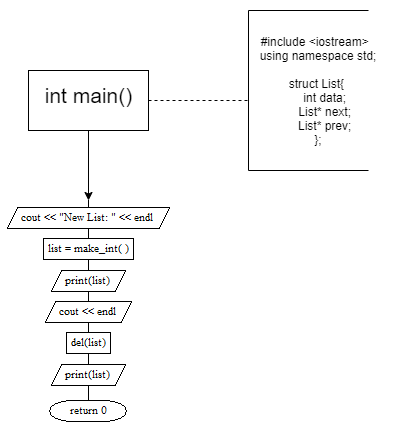
1. Объявляем структуру List с информационным полем int.
2. Помимо указателя на следующий элемент, добавляем указатель на предыдущий.
3. В функции печати с помощью цикла печатаем все элементы, пока указатель не равен NULL, постепенно переходя от первого элемента к последнему.
4. Используются переменные типа int и структура List с информационным полем int. Указатели на следующий элемент List\* next и на предыдущий List\* prev.
5. В программе используются циклы while, for.

**Блок-схема:**

****







**Код программы:**

#include <iostream>

using namespace std;

struct List{

    int data;

    List\* next;

    List\* prev;

};

List\* make\_int()

{

    List\* first, \*p;

    first = NULL;

    p = new List;

    p->data = rand() %100;

    p-> prev = NULL;

    p-> next =NULL;

    first = p;

    for (int i = 1; i<5; i++)

    {

        List\* h = new List;

        List\* q = p;

        p->next = h;

        p = p->next;

        p->data = rand() %100;

        p->next = NULL;

        p->prev = q;

    }

     return first;

}

void print(List\* first)

{

    if( first == NULL)

    {

        cout << "Empty lIst";

    }

    List\* p= first;

    while (p!=NULL)

    {

        cout << p->data << endl;

        p=p->next;

    }

    cout << endl;

}

void del(List\* first)

{

    cout << "Delete all ODD" << endl;

    if (first == NULL)

    {

        cout << "Empty lIst" << endl;

    }

    List\* p = first;

    List\* q = NULL;

        if ( p-> data %2 == 0)

        {

            q=p;

        }

        p=p->next;

        while (p->next !=NULL)

        {

            if (p->next->data %2 == 00)

            {

                q=p;

            }

            p=p->next;

        }

        if (q != NULL)

        {

            p=q;

            p->next=p->next->next;

            delete q;

        }

}

int main()

{

    cout << "New List: " <<  endl;

    List\* list= make\_int();

    print (list);

    cout << endl;

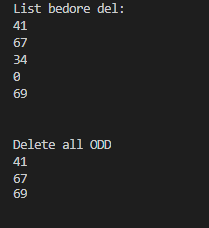
    del (list);

    print (list);

    return 0;

}

**Результат работы программы:**

****