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 11.3**

Дисциплина: Основы алгоритмизации и программирования

Выполнил:

Студентка группы ИВТ-22-2б

Мифтахов Марат Ринатович

Проверил:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь, 2023**

**Постановка задачи**

Тип информационного поля char\*. Добавить в стек элементы с номерами 1, 3, 5 и т. д.

**Анализ задачи**

1. Создаём структуру стека Stack

struct Stack

{

char data;

Stack\* prev;

};

1. Создаём функцию для заполнения стека

Stack\* Create(int n)

{

if (n == 0)

{

return NULL;

}

Stack\* top, \* p;

top = NULL;

p = new Stack;

cout << "Введите элементы: ";

cin >> el;

p->data = el;

p->prev = NULL;

top = p;

for (int i = 2; i <= n; i++)

{

Stack\* h = new Stack;

cin >> el;

h->data = el;

h->prev = top;

top = h;

}

return top;

}

1. Создаём функцию для добавления элемента с нечётным номером в стек

void Add(Stack\*& first, int n)

{

Stack\* p = first;

Stack\* h = new Stack;

cout << "Введите элементы для добавления: ";

cin >> h->data;

h->prev = p;

first = h;

while (p->prev != NULL)

{

Stack\* h = new Stack;

cin >> h->data;

h->prev = p->prev;

p->prev = h;

if (p->prev->prev != NULL)

{

p = p->prev->prev;

}

else

{

p = p->prev;

}

}

}

1. Для вывода создаём функцию Print

void Print(Stack\* top)

{

if (top == NULL)

{

cout << "Стек пуст" << endl;

}

else

{

Stack\* p = top;

while (p != NULL)

{

cout << p->data << " ";

p = p->prev;

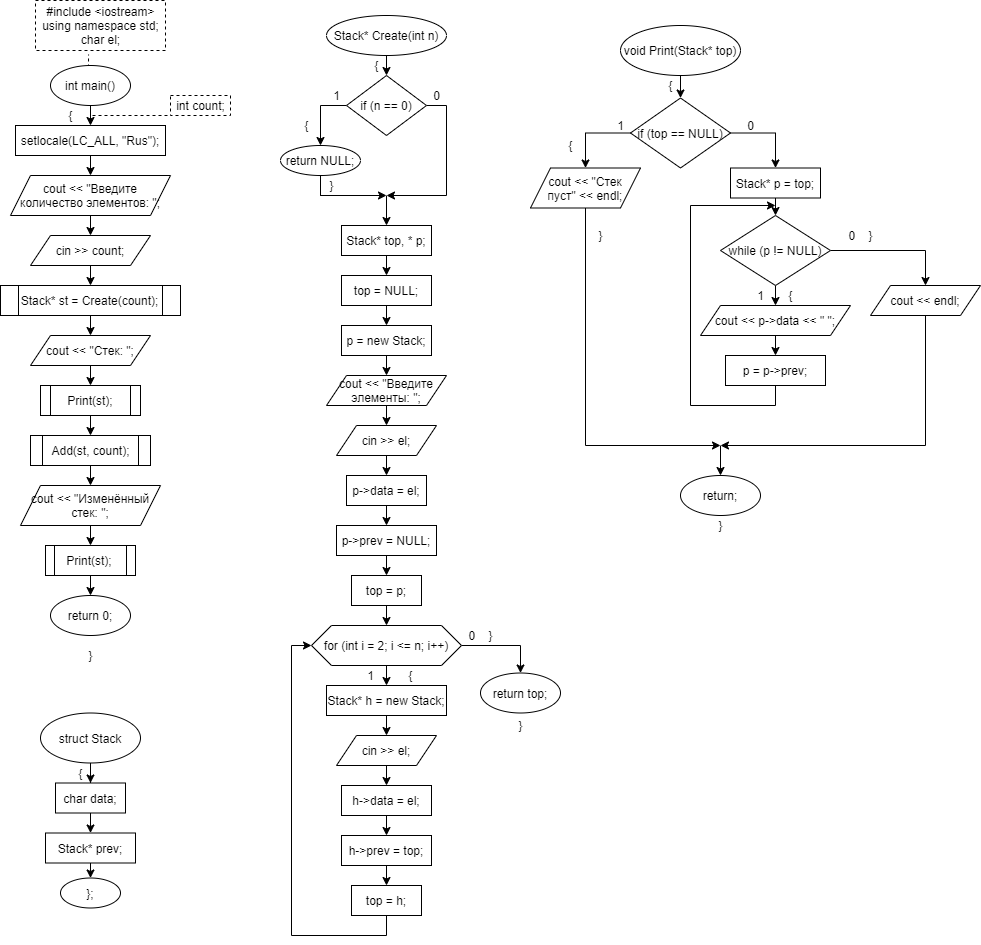
}

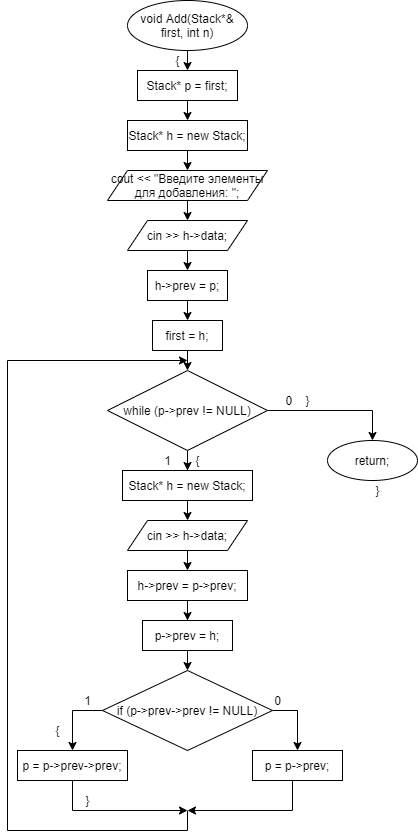
cout << endl;

}

}

**Блок-схема**



**Исходный код**

#include <iostream>

using namespace std;

char el;

struct Stack

{

char data;

Stack\* prev;

};

Stack\* Create(int n)

{

if (n == 0)

{

return NULL;

}

Stack\* top, \* p;

top = NULL;

p = new Stack;

cout << "Введите элементы: ";

cin >> el;

p->data = el;

p->prev = NULL;

top = p;

for (int i = 2; i <= n; i++)

{

Stack\* h = new Stack;

cin >> el;

h->data = el;

h->prev = top;

top = h;

}

return top;

}

void Print(Stack\* top)

{

if (top == NULL)

{

cout << "Стек пуст" << endl;

}

else

{

Stack\* p = top;

while (p != NULL)

{

cout << p->data << " ";

p = p->prev;

}

cout << endl;

}

}

void Add(Stack\*& first, int n)

{

Stack\* p = first;

Stack\* h = new Stack;

cout << "Введите элементы для добавления: ";

cin >> h->data;

h->prev = p;

first = h;

while (p->prev != NULL)

{

Stack\* h = new Stack;

cin >> h->data;

h->prev = p->prev;

p->prev = h;

if (p->prev->prev != NULL)

{

p = p->prev->prev;

}

else

{

p = p->prev;

}

}

}

int main()

{

setlocale(LC\_ALL, "Rus");

cout << "Введите количество элементов: ";

int count;

cin >> count;

Stack\* st = Create(count);

cout << "Стек: ";

Print(st);

Add(st, count);

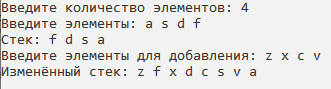
cout << "Изменённый стек: ";

Print(st);

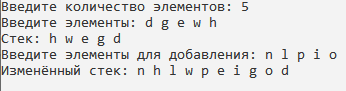
return 0;

}

**Скриншоты программы**

****

****

****

**Анализ (объяснение скриншотов)**

Программа запрашивает у пользователя количество элементов стека и все элементы стека, затем выводит получившийся стек. Далее программа запрашивает у пользователя элементы, которые нужно добавить в стек, и выводит измененный стек.