Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное учреждение высшего образования «Пермский национальный исследовательский политехнический университет»

ПНИПУ

**Лабораторная работа №11. Стеки**

Выполнил:   
студент группы РИС-23-1б   
Сингур Иван Сергеевич

Проверила:   
доцент кафедры ИТАС   
О.А. Полякова

Пермь, 2024 г.

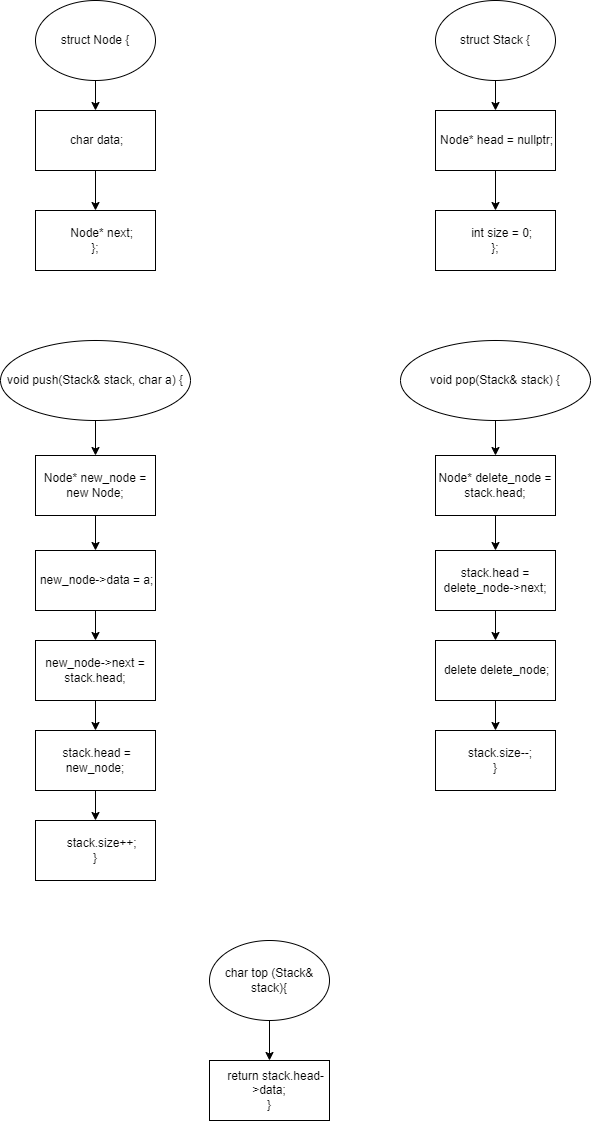
**Условие:**

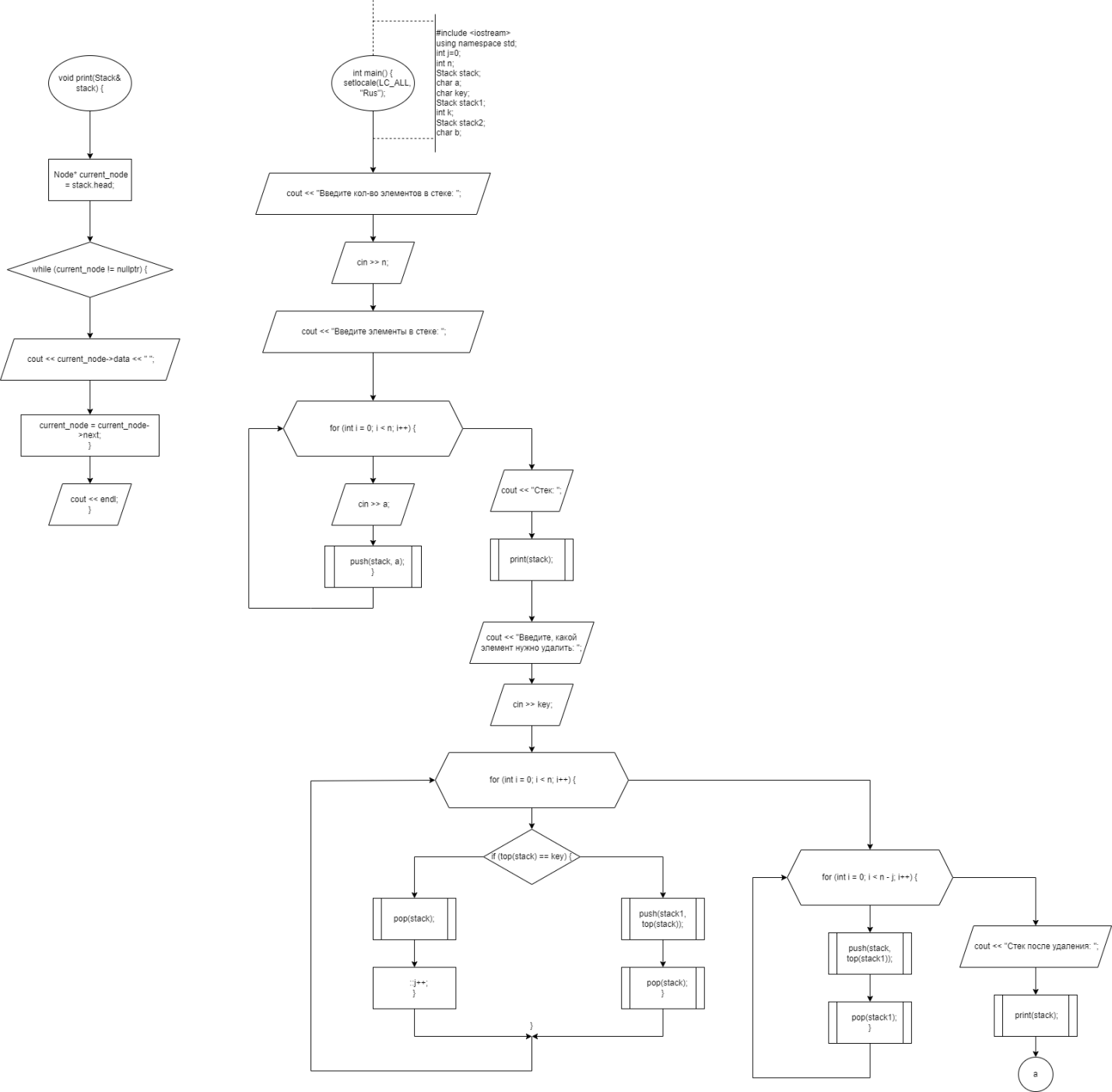
Cтек должен содержать ключевое поле типа \*char (строка символов). Сформировать стек. Удалить элемент с заданным ключом. Добавить по К элементов в начало и в конец списка.

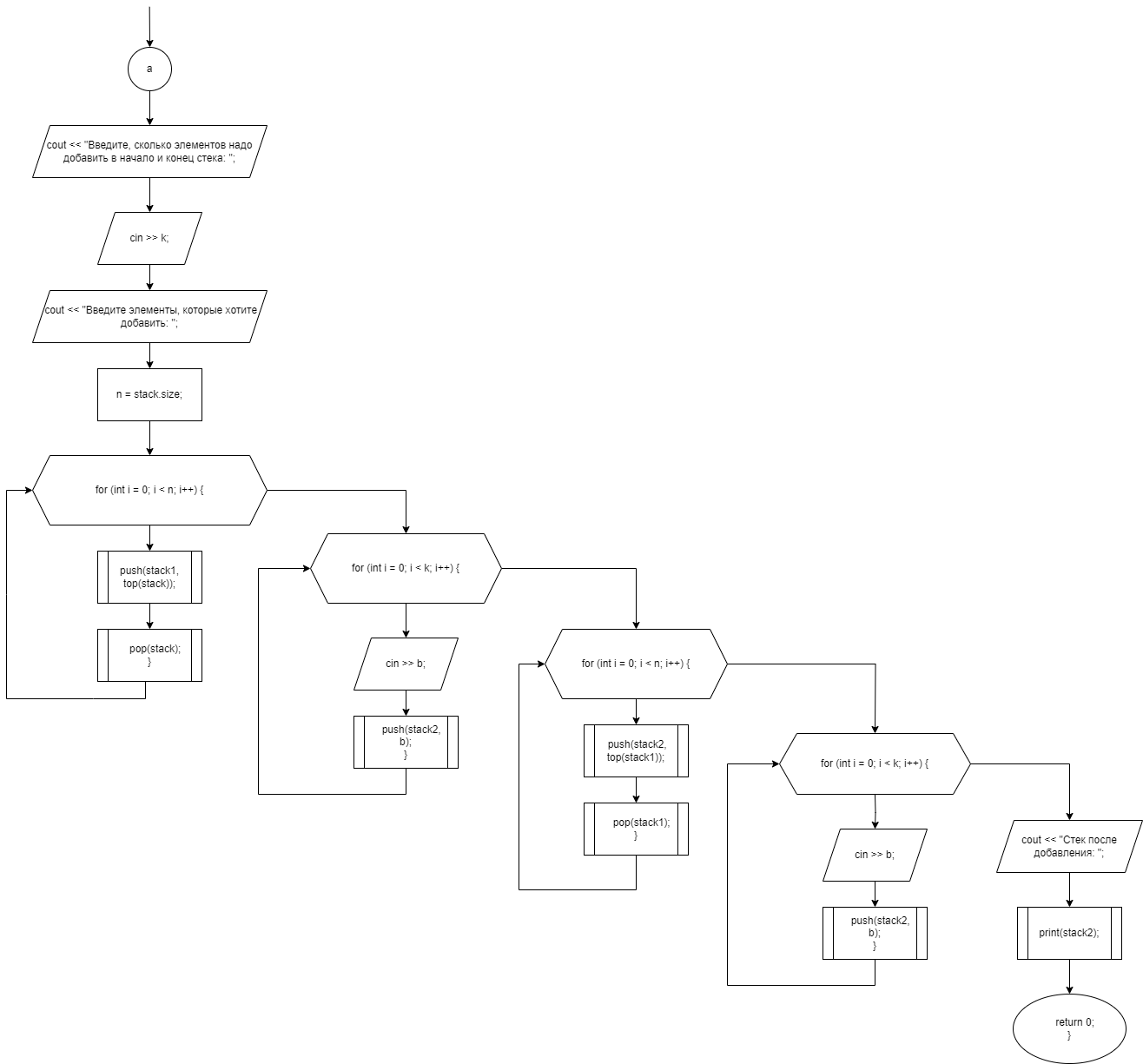
**Анализ задачи:**

1. Первым делом создаём стек и заполняем его функцией push, параметрами которой будут являться сам стек, который мы будем заполнять и данные (строка символов), которые мы будем вводить.
2. С помощью функции print выводим полученный стек на экран.
3. Для того, чтобы удалить элемент в стеке, создаём ещё один, в который мы переносим оставшиеся элементы, кроме ключевого элемента.
4. Выводим полученный стек на экран с помощью функции print.
5. Создаем переменную k, которая будет означать, сколько элементов мы будем добавлять.
6. С помощью циклов и функций push и pop создаем получившийся стек с новыми элементами в начале и конце.
7. Выводим его printom на экран

**Блок-схема:**

****

****

****

**Код на языке C++:**

#include <iostream>

using namespace std;

int j;

struct Node {

char data;

Node\* next;

};

struct Stack {

Node\* head = nullptr;

int size = 0;

};

void push(Stack& stack, char a) {

Node\* new\_node = new Node;

new\_node->data = a;

new\_node->next = stack.head;

stack.head = new\_node;

stack.size++;

}

void pop(Stack& stack) {

Node\* delete\_node = stack.head;

stack.head = delete\_node->next;

delete delete\_node;

stack.size--;

}

void print(Stack& stack) {

Node\* current\_node = stack.head;

while (current\_node != nullptr) {

cout << current\_node->data << " ";

current\_node = current\_node->next;

}

cout << endl;

}

char top (Stack& stack){

return stack.head->data;

}

int main() {

setlocale(LC\_ALL, "Rus");

int n;

Stack stack;

char a;

cout << "Введите кол-во элементов в стеке: ";

cin >> n;

cout << "Введите элементы в стеке: ";

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

cin >> a;

push(stack, a);

}

cout << "Стек: ";

print(stack);

char key;

cout << "Введите, какой элемент нужно удалить: ";

cin >> key;

Stack stack1;

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

if (top(stack) == key) {

pop(stack);

::j++;

}

else {

push(stack1, top(stack));

pop(stack);

}

}

for (int i = 0; i < n - j; i++) {

push(stack, top(stack1));

pop(stack1);

}

cout << "Стек после удаления: ";

print(stack);

int k;

cout << "Введите, сколько элементов надо добавить в начало и конец стека: ";

cin >> k;

Stack stack2;

char b;

cout << "Введите элементы, которые хотите добавить: ";

n = stack.size;

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

push(stack1, top(stack));

pop(stack);

}

for (int i = 0; i < k; i++) {

cin >> b;

push(stack2, b);

}

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

push(stack2, top(stack1));

pop(stack1);

}

for (int i = 0; i < k; i++) {

cin >> b;

push(stack2, b);

}

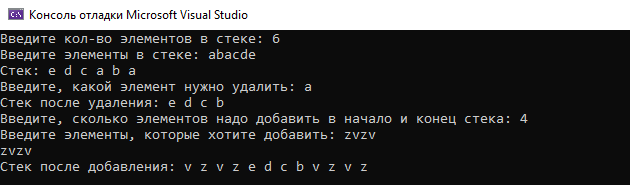
cout << "Стек после добавления: ";

print(stack2);

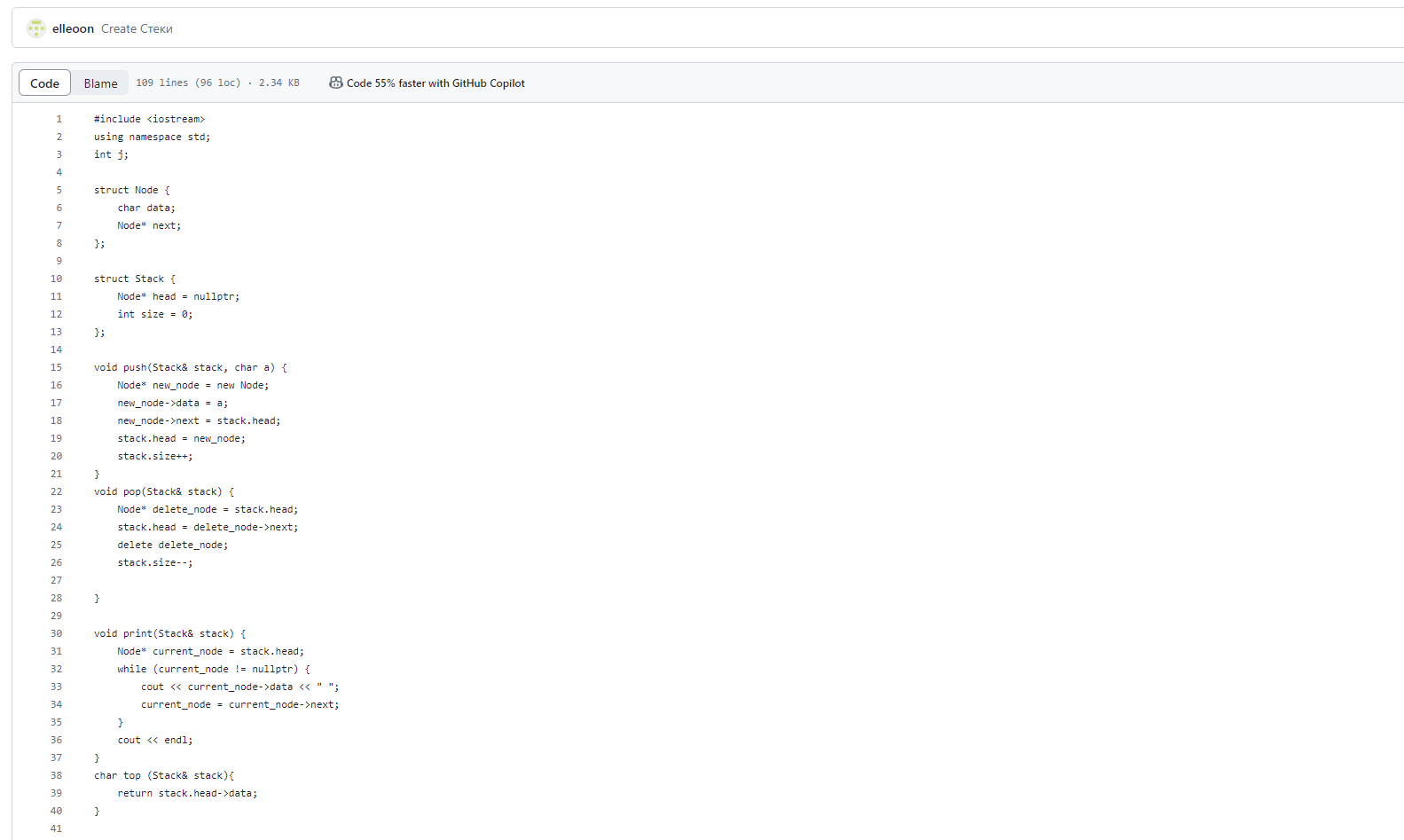
return 0;

}

**Работа программы:**



**Cкрины из гита:**



**Ссылка на гит:**

https://github.com/elleoon/pnipu1

**Вывод:** Задача была выполнена. Всё получилось.