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»

Тема: Информационные и динамические структуры. Стэки.

Вариант 4

Выполнил:

Студент группы РИС-20-1б

Еске Вячеслав Сергеевич

Проверила:

Доцент кафедры ИТАС

Полякова О. А.

**Пермь, 2021**

**Цель работы**

Получить практические навыки работы со стеками.

**Постановка задачи**

1. Тип информационного поля int. Удалить из стека последний элемент с четным информационным полем.

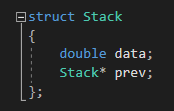
2. Распечатать полученную структуру.

3. Выполнить обработку структуры в соответствии с заданием.

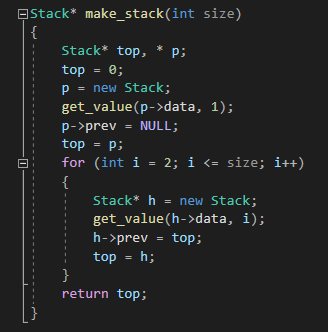
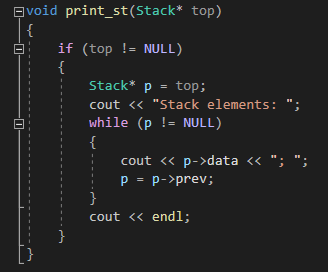
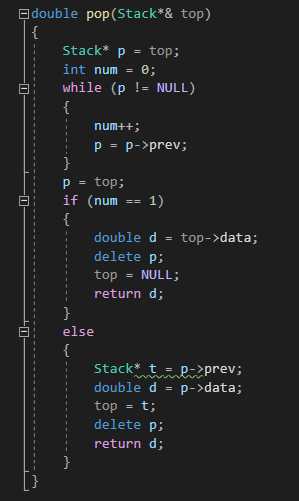
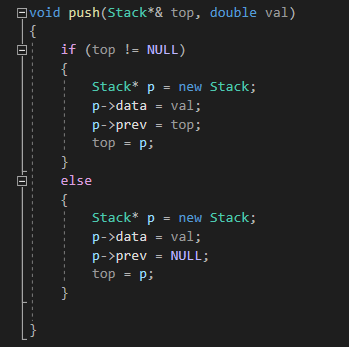
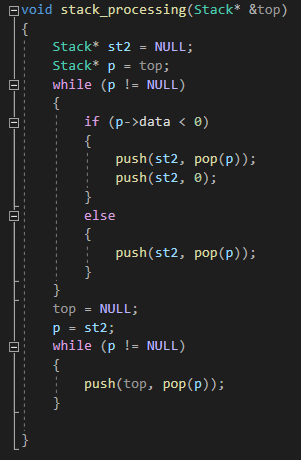
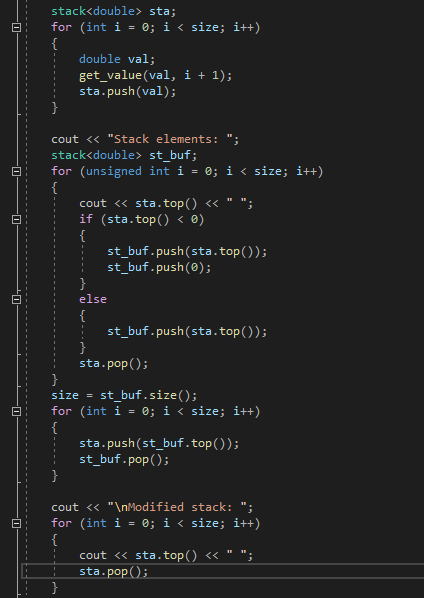
4. Распечатать полученный результат.

5. Удалить соответствующую структуру из памяти.

**Анализ задачи**

1. Определить какие действия необходимо выполнить для решения задачи.
   1. Реализация стэка производится через структуру Stack. 
   2. Создать стэк.



* 1. Заполнить стэк вещественными числами. 
  2. Вывести все элементы стэка в консоль. 
  3. Удаление первого элемента стэка. 
  4. Добавление элемента в стэк. 
  5. Поиск отрицательных значений в стэке и добавление 0. 
  6. Реализация через STL. 

1. Для решения задачи использовались следующие типы данных.

2.1 Данные элементов списка хранятся в формате int.



2.2 Указатель на следующий элемент является экземпляром структуры.



**Код**

#include <iostream>

using namespace std;

int a;

bool f=true;

struct Stack {

int data;

Stack\* prev;

};

Stack\* x;

Stack\* make\_stack(int n) {

if (n == 0) return NULL;

Stack\* top, \* p;

top = NULL;

p = new Stack;

cin >> a;

p->data = a;

if ((p->data % 2 == 0) && f) {

x = p;

f = false;

}

p->prev = NULL;

top = p;

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

Stack\* h = new Stack;

cin >> a;

h->data = a;

if ((h->data % 2 == 0) && f) {

x = h;

f = false;

}

h->prev = top;

top = h;

}

return top;

}

void print\_stack(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 delete\_stack(Stack\*& top) {

Stack\* p = top;

if (p == x) {

p = p->prev;

delete top;

top = p;

}

else {

while ((p != NULL)) {

if (p->prev == x) {

Stack\* q = p->prev;

p->prev = p->prev->prev;

delete q;

}

else p = p->prev;

}

}

}

int main()

{

setlocale(LC\_ALL, "rus");

int n;

do {

cout << "Введите количество элементов: ";

cin >> n;

} while (n <= 0);

cout << "Введите элементы: ";

Stack\* st = make\_stack(n);

cout << "Стек: ";

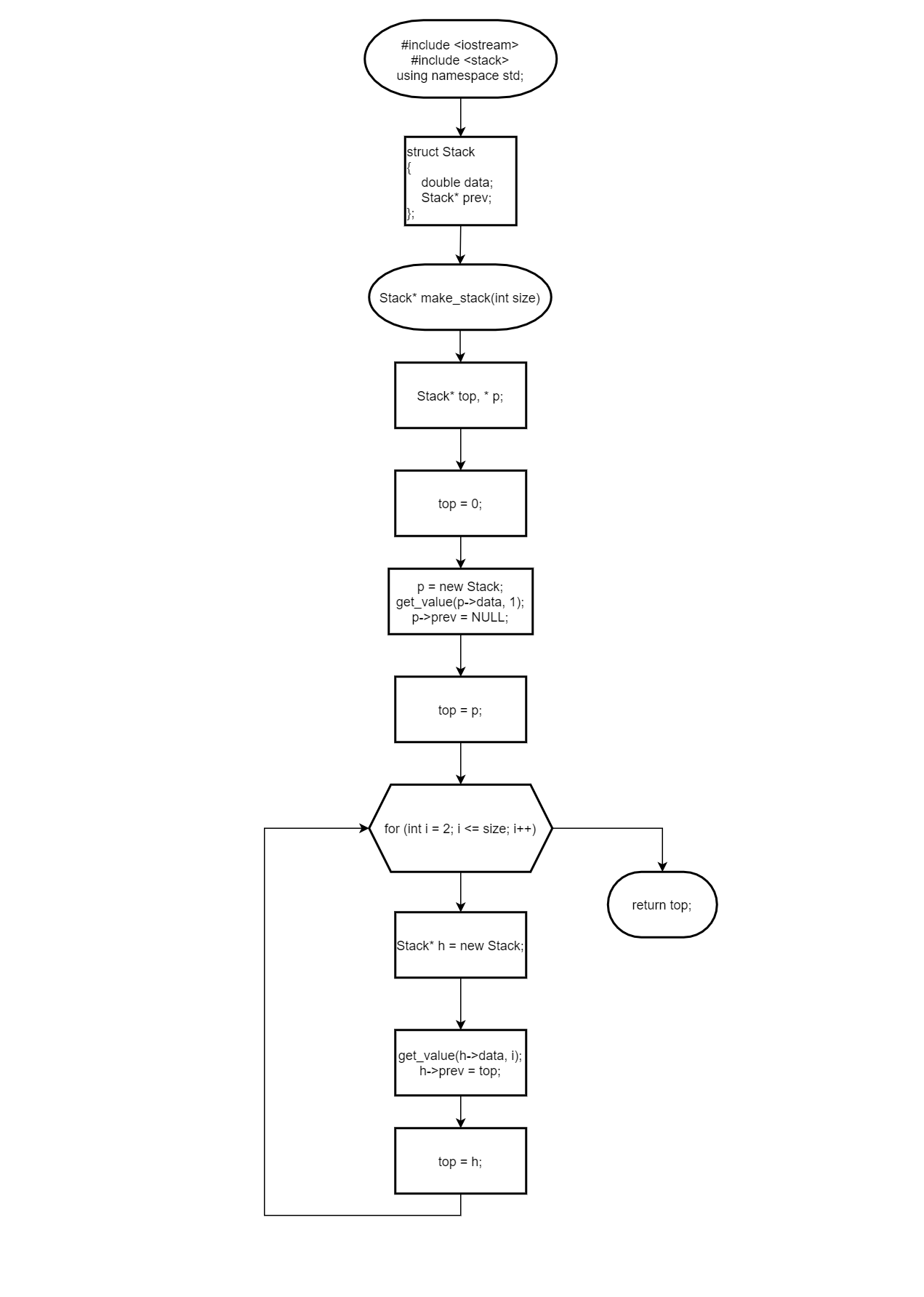
print\_stack(st);

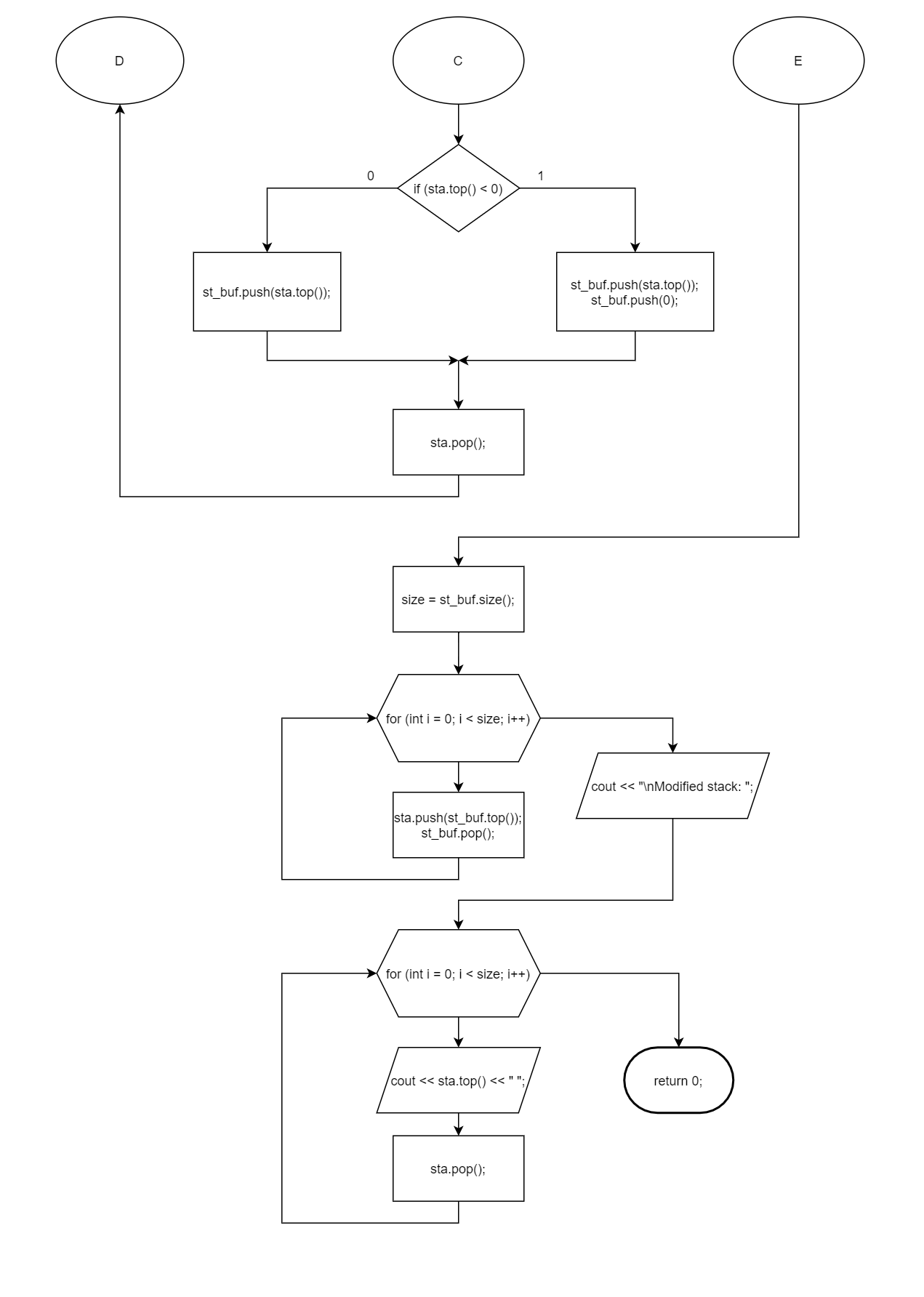
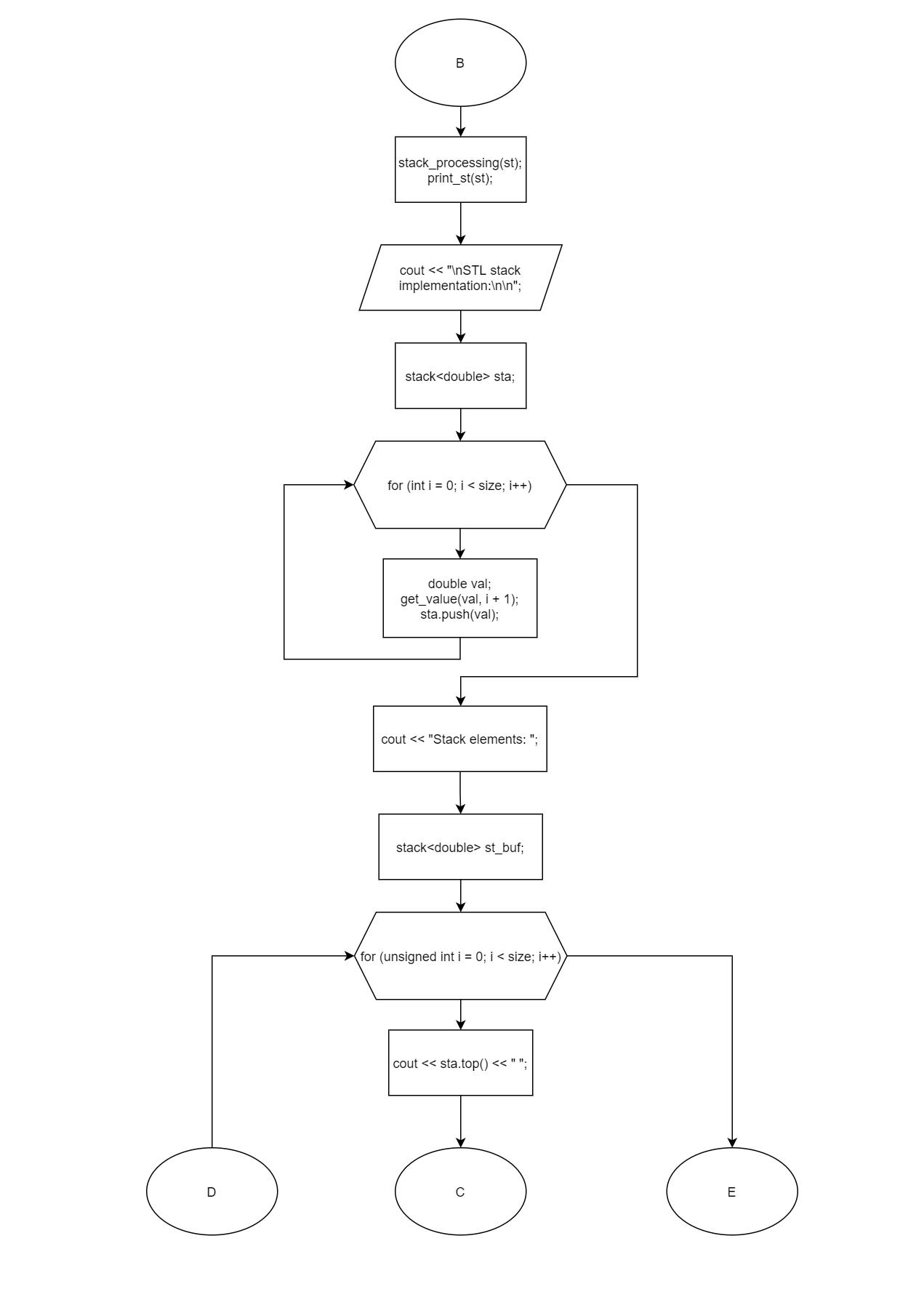
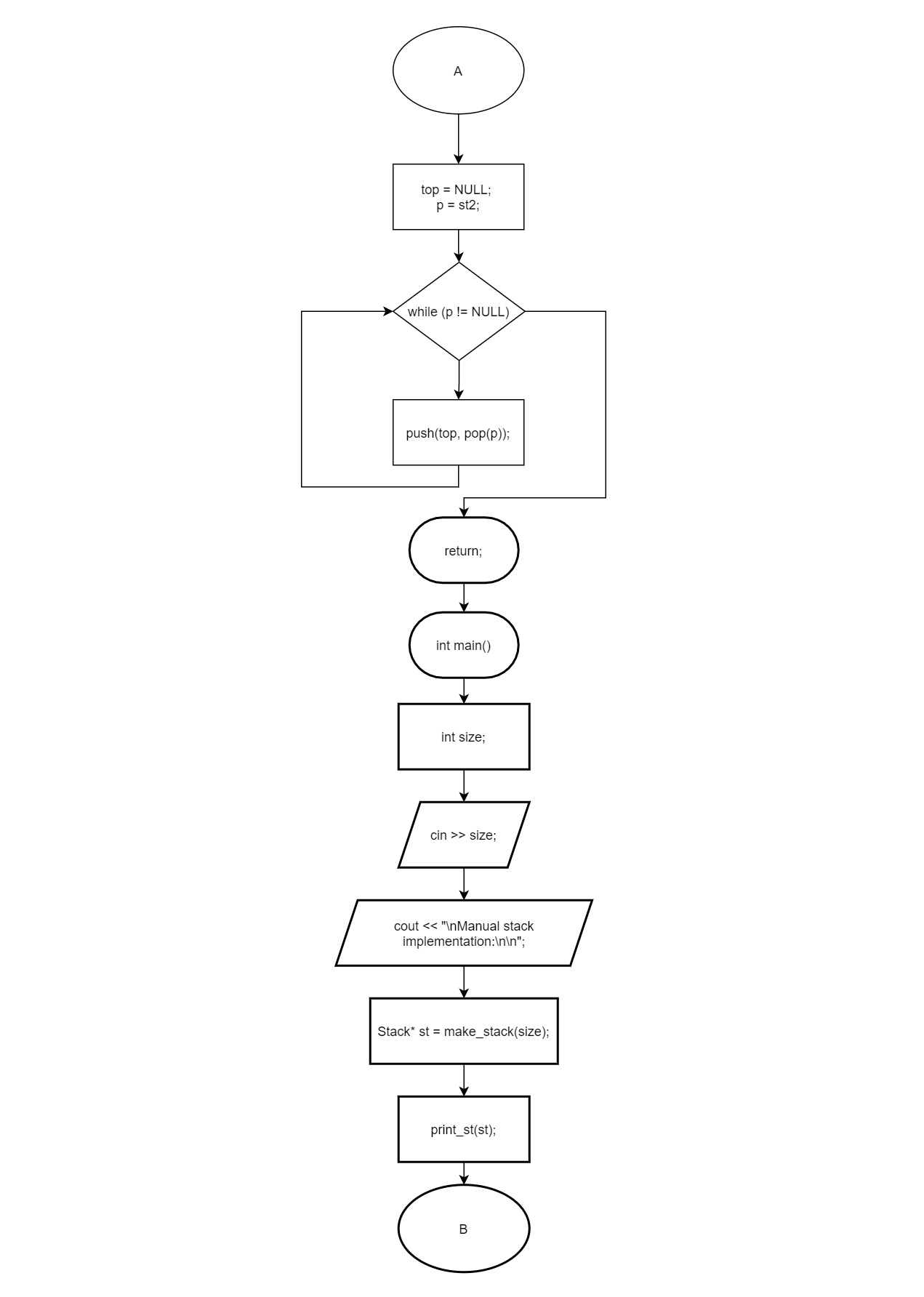
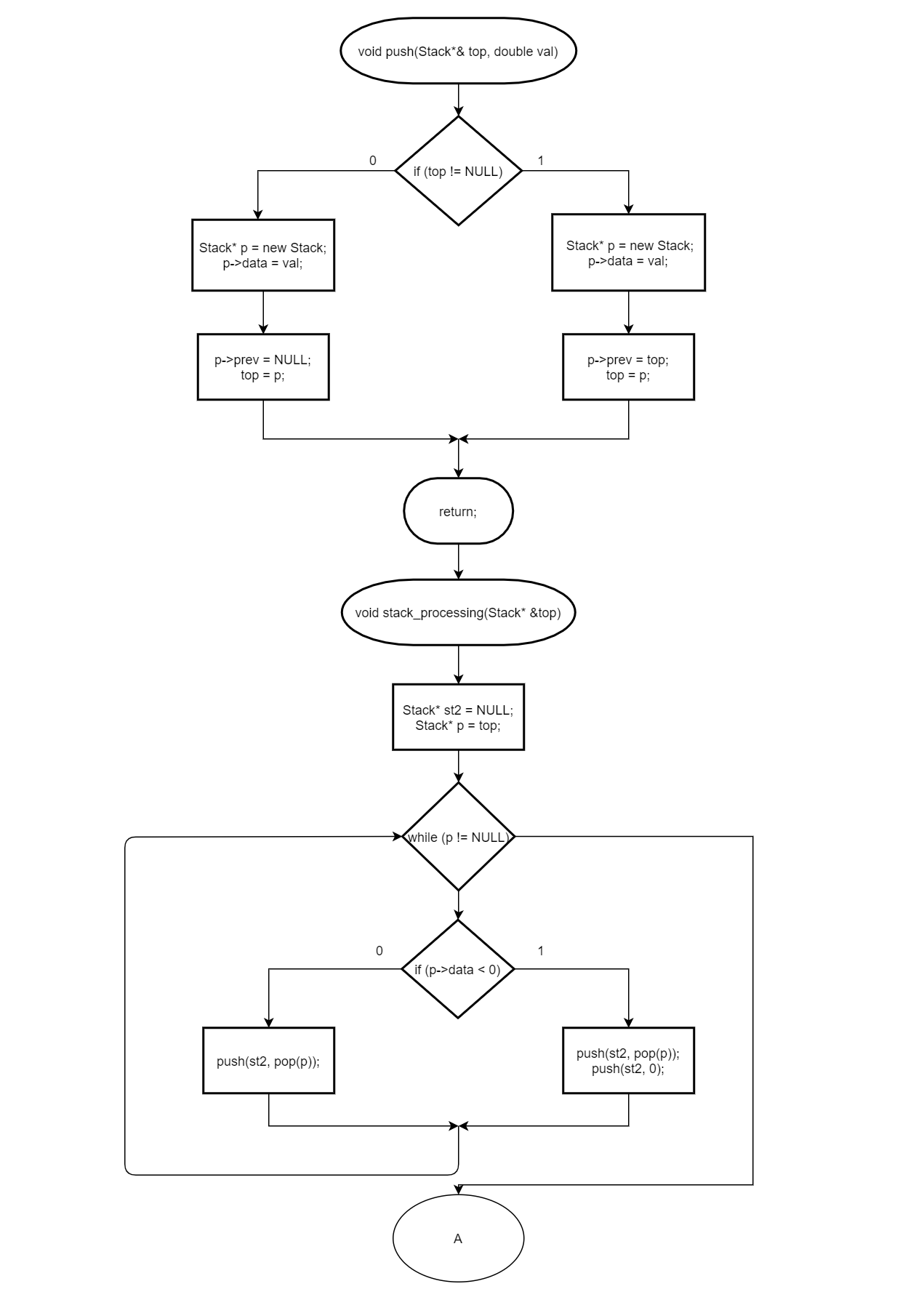
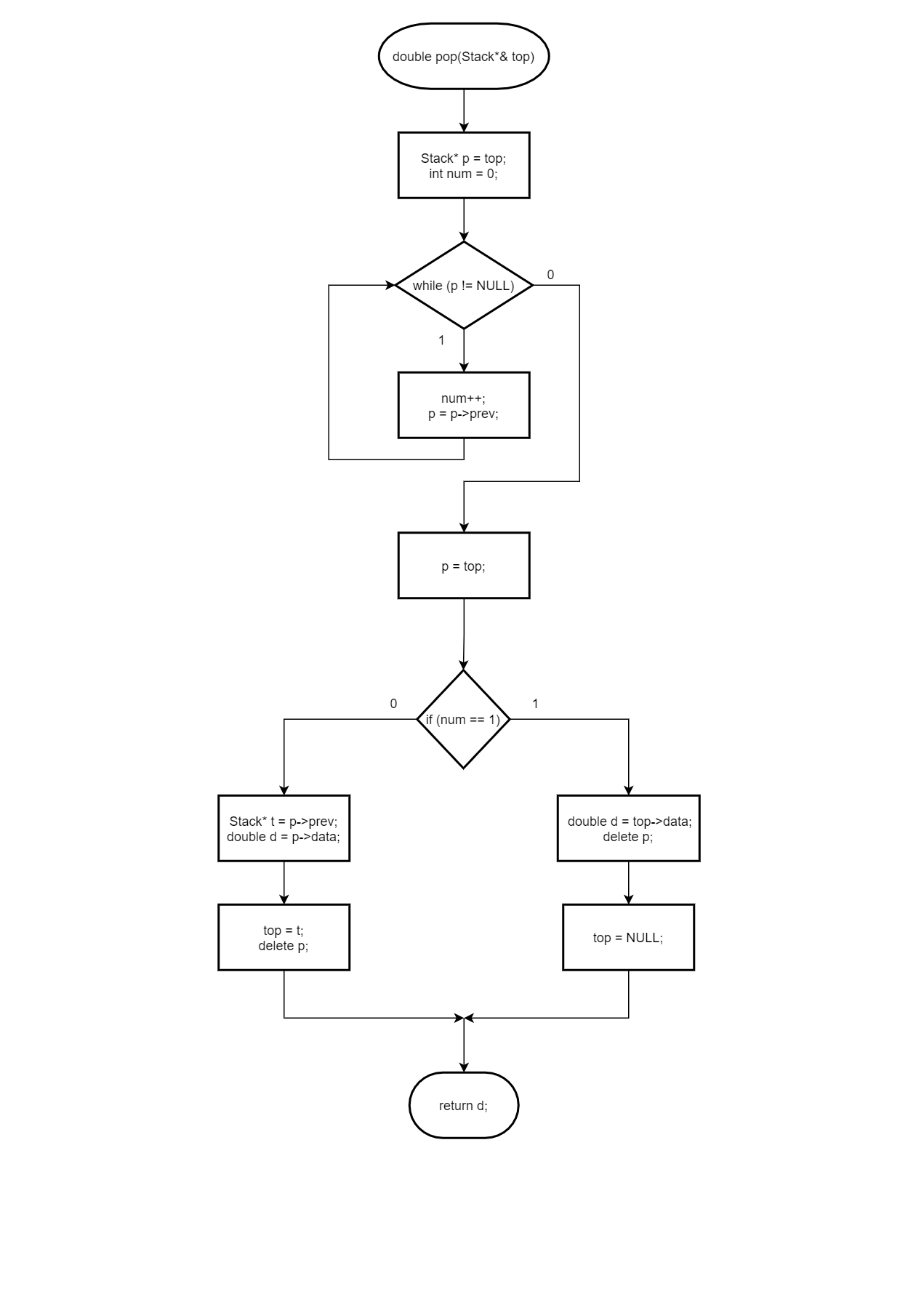
delete\_stack(st);

cout << "Стек: ";

print\_stack(st);

}

**Блок схема **

****

**Тесты**

