Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«Пермский национальный исследовательский политехнический университет» (ПНИПУ)

Электротехнический факультет

Кафедра информационных технологий и автоматизированных систем

Направление подготовки 09.03.01 «Информатика и вычислительная техника»

**Лабораторная работа № 11.3 (две реализации)**

**по дисциплине**

**«Основы алгоритмизации и программирования»**

**семестр** 2

Выполнил студент гр. ИВТ-21-1б

Балдин Алексей Игоревич

Проверил:

Яруллин Денис Владимирович

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

(оценка) (подпись)

\_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

г. Пермь - 2022

**Цель**

Получить практические навыки работы со стеками. Стек – это частный случай однонаправленного списка.

**Постановка задачи**

Сформировать стек. Тип информационного поля указан в варианте. Распечатать полученную структуру. Выполнить обработку структуры в соответствии с заданием. Распечатать полученный результат.

**Исходные данные**

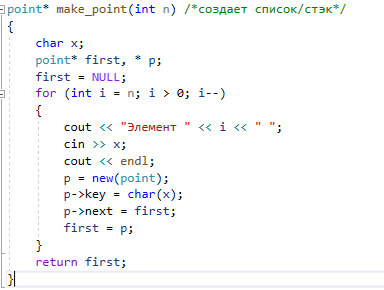
Вариант 5:

Сначала представлю реализацию и блок-схему к стеку (БЕЗ библиотеки), после, будет стек (С ПОМОЩЬЮ библиотеки).

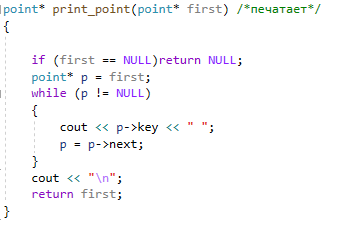
Тип информационного поля char. Добавить в стек элемент после элемента с заданным информационным полем.

**Итак, реализация стека (без библиотеки):**

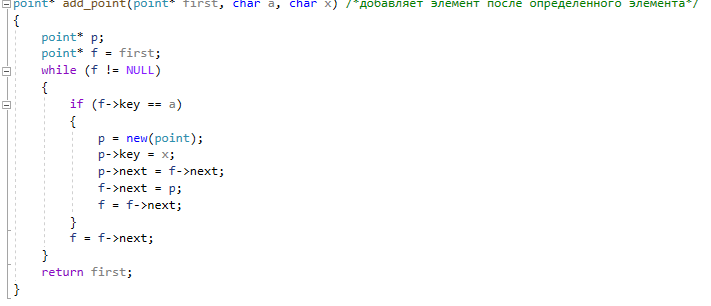
Данная функция создает стек, то есть в ней ввод элементов, каждому элементу присваиваются данные (p -> key = char(x)), и указатель на следующий элемент ставлю на предыдущий элемент, так, чтобы они вывелись в обратном порядке.



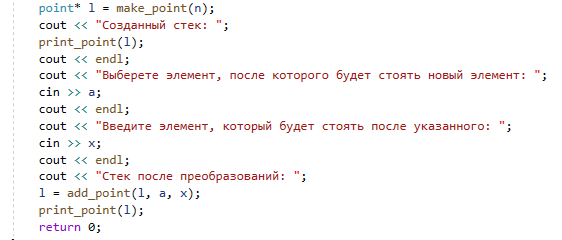
Функция выводит введенный стек на экран, при условии, что указатель на вершину стека не нуль, иначе возвращаем нуль. В цикле не забываем переместить указатель на следующий. Функция возвращает указатель на вершину стека.



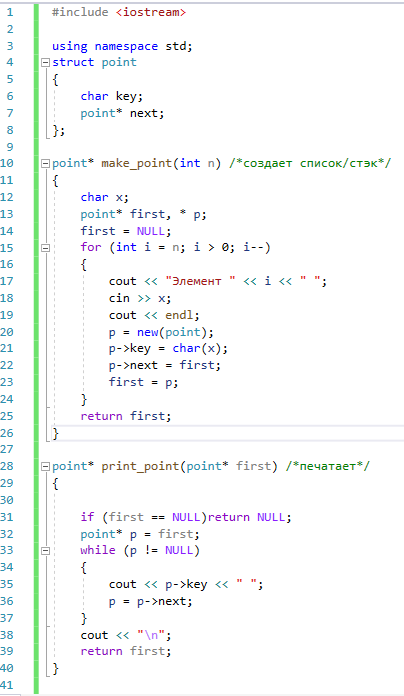
Данная функция добавляет элементы после указанного элемента. Создаем вспомогательный указатель (p) и (f) – дубликат указателя на вершину. В цикле пока указатель на вершину не равен нулю, переходим (if), и сравниваем элемент стека с введённым символом, если равны, то создаем элемент стека, присваиваем туда новый элемент, указатели распределяем, тем самым вставляем элемент после выбранного.



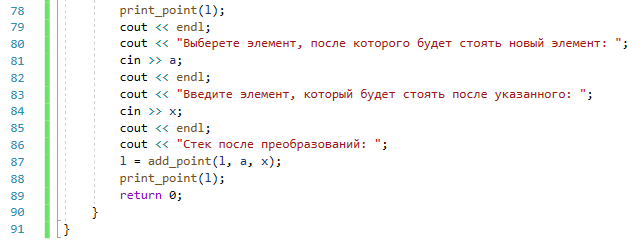
В (main) через дополнительную структуру обращаюсь ко всем функциям, и вывожу результат на экран.



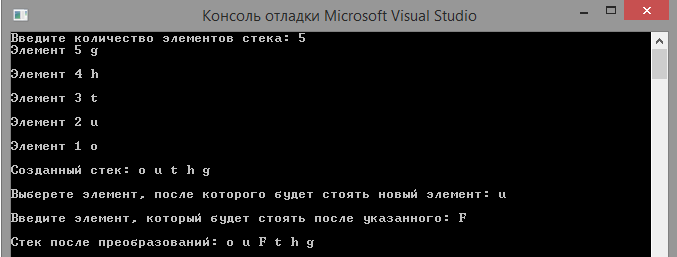
Код программы:







Результат работы программы:



Результат 1

Блок-схема к программе:

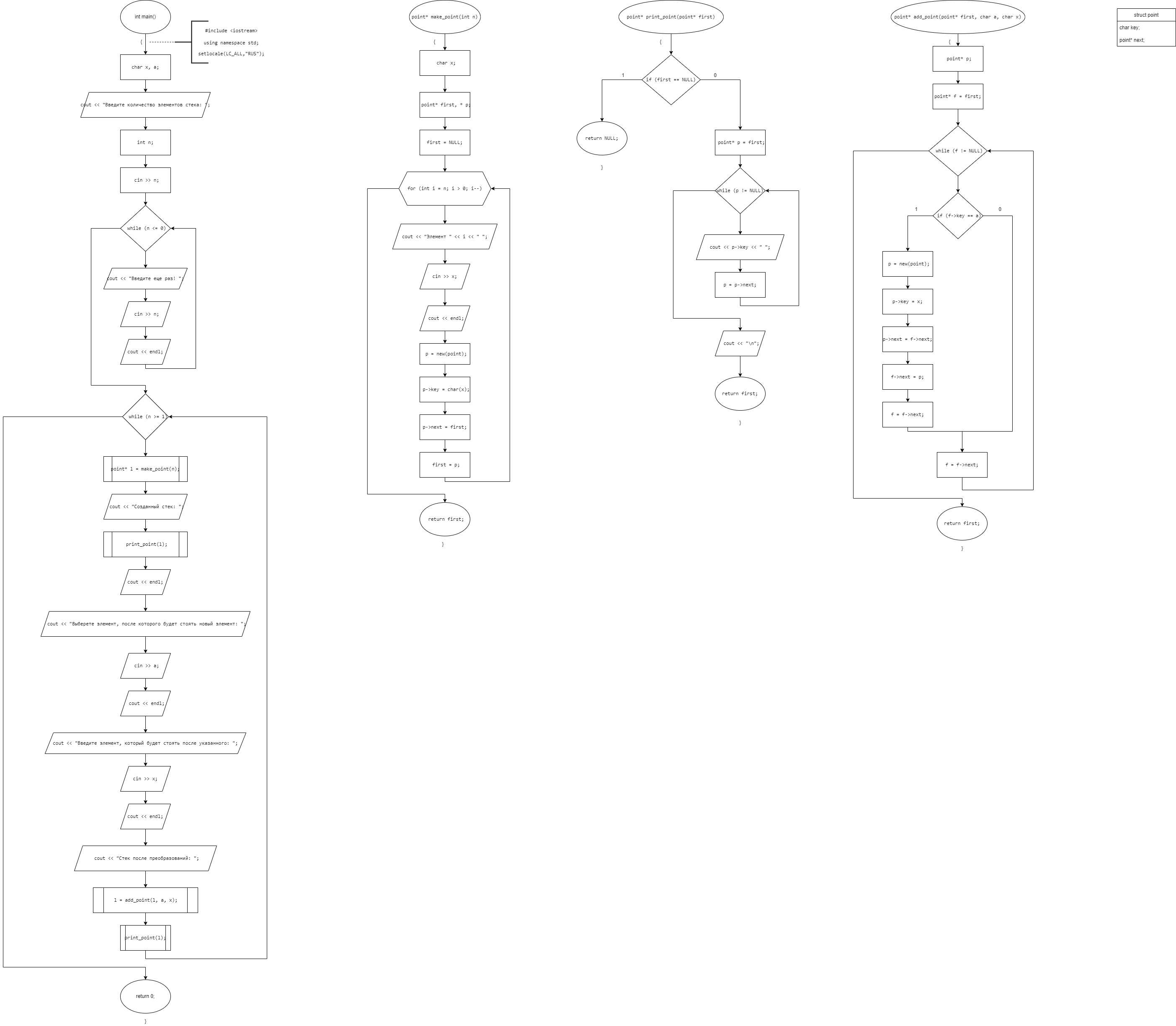


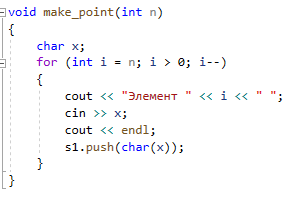
Рисунок 1

**Итак, реализация стека (С помощью библиотеки):**

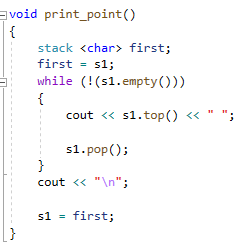
Когда мы работаем с библиотекой, стек создаётся таким образом:



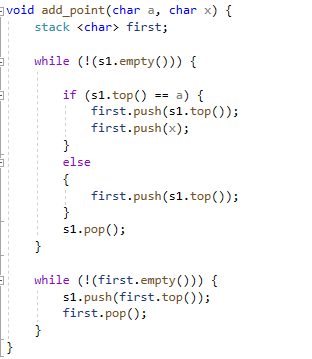
Здесь функция создает стек, как и в прошлой реализации, просто теперь мы пользуемся указателями не в ручную, а с помощью встроенных функций библиотеки «stack», а именно s1.push – значит заполняем стек элементами, которые мы ввели.



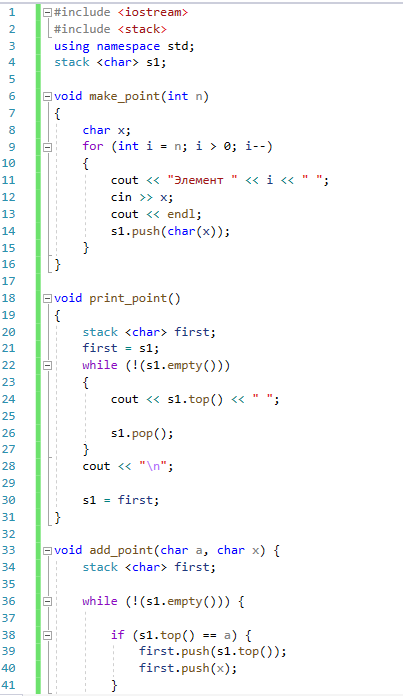
Данная функция выводит стек на экран, при условии, что вершина стека не пуста, после вывода на экран вершины, мы её удаляем, с помощью функции (pop), тем самым наша вершина становится на элемент ниже, и так происходит в цикле, в результате выводим элементы стека на экран.

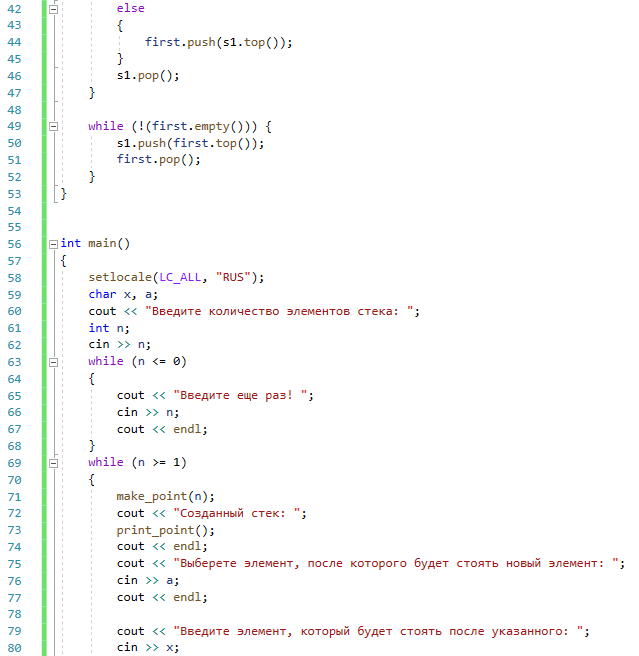


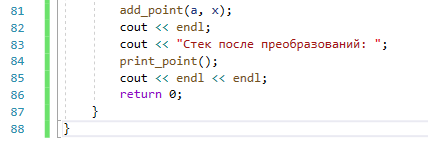
Здесь функция с параметрами символов, один из которых представляет собой элемент из стека, а другой, новый элемент, который будет стоять после первого. Пока вершина не пуста, сравниваем вершину и выбранный элемент, если равны, то вносим новый элемент выше вершины, после ее удаляем. Но далее мы ее возвращаем в следующем цикле while.



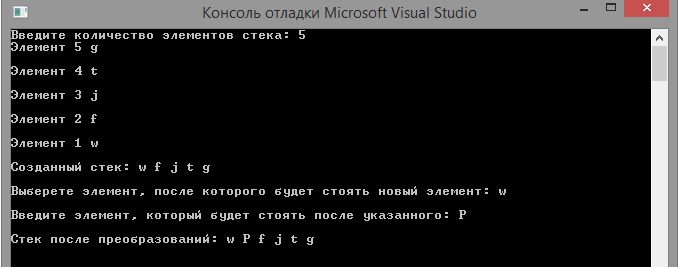
Код программы:







Результат работы программы:



Результат 2

Блок-схема к задаче:

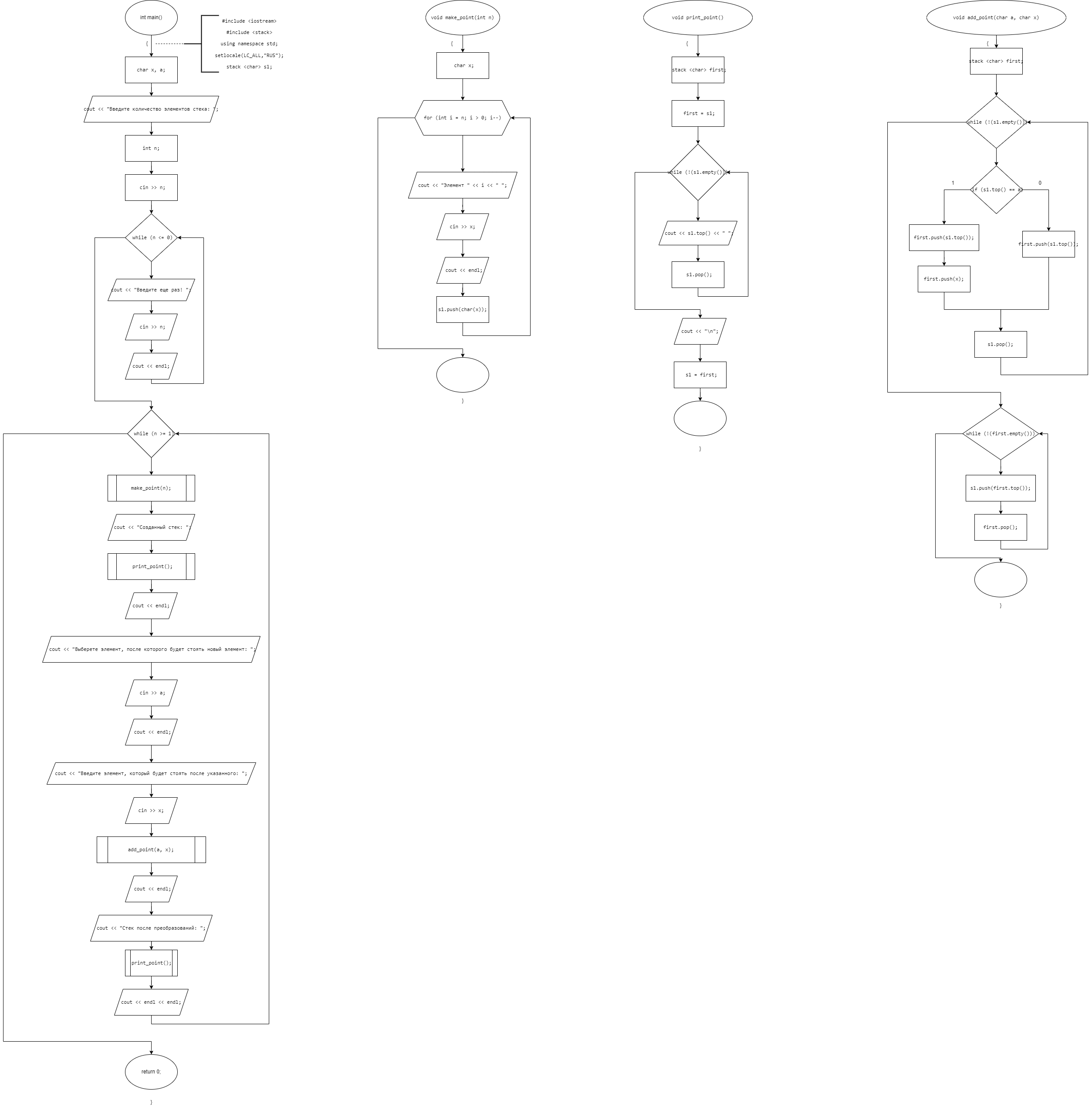


Рисунок 2