Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное учреждение высшего образования

«Пермский национальный исследовательский политехнический университет»

ПНИПУ

**Лабораторная работа**

**“Сдвиг массива ”**

Выполнил:

студент группы РИС-23-2б

Швацкий Даниил Александрович.

Проверила:

доцент кафедры ИТАС

О.А. Полякова

2024 г.

**Постановка задачи:** 1) Задан одномерный массив. Перед каждым четным значением элемента поставить -1. В новом массиве удалить все нечетные элементы. 2) Задан одномерный массив. Проверить, упорядочен ли массив по возрастанию.

1) Объявляем массив int arr[100] на 100 элементов, и переменную int n, отвечающую за границу массива.

С помощью цикла do while (n < 2 || n > 100) делаем проверку, чтобы значение n не выходило за границу массива. И просим пользователя ввести n (cin >> n)

Заполняем массив с помощью цикла for (int i = 0; i < n; ++i), используем генератор случайных чисел (arr[i] = rand % 100)

Циклом for (int i = 0; i < n; ++i) просматриваем каждый элемент массива. На каждой итерации делаем проверку, если элемент с четным значением, то расширяем границу массива (n++), далее циклом for (int j = n; j > i; --j) сдвигаем все элементы вправо (arr[j] = arr[j - 1]), после чего заменяем текущий элемент на arr[i] = -1, сдвигаем итерацию (i++), чтобы снова не проходить по тому же элементу.

Выводим получившийся массив на экран.

Воспользуемся циклом for (int i = 0; i < n; ++i), чтобы просматривать каждый элемент массива. Если элемент является нечетным значением, то уменьшем границу массива (n--), и сдвигаем на этот элемент остальные элементы массива (arr[j] = arr[j + 1]) с помощью цикла for (int j = i; j < n; ++j). Сдвигаем итерацию (i--), чтобы проверить на четность новый arr[i] элемент.

Выводим массив на экран.

2) Подключим директиву clocale. В int main() пропишем функцию setlocale(LC\_ALL, “ru”) для корректного вывода русского языка. Объявим массив int arr[100] на 100 элементов, переменную int n, отвечающую за количество элементов массива, с которыми будем взаимодействовать; переменная int temp нужна для инициализации массива числами введенными пользователем.

Циклом do while(n < 2 || n > 100) проврим не выходит ли n за границы массива.

Инициализируем переменную int j =0, отвечающую за количество элементов которыми будет заполняться массив. Циклом do while(j < n) просим пользователя ввести элементы с клавиатуры.

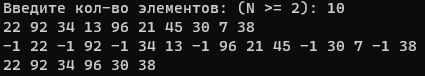
Обнулим переменную j, инициализируем переменную bool flag = true, которая будет отвечать за проверку на упорядоченность.

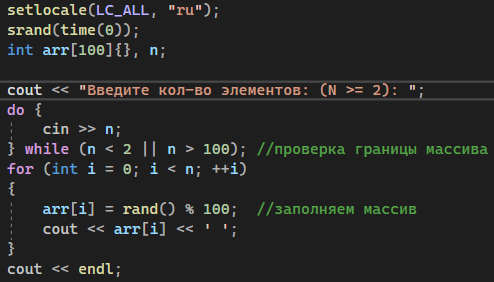
Циклом while(flag && j < n - 1) будем проверять выполнение условия на каждой итерации до тех пор пока не проверим весь массив, или flag = false.

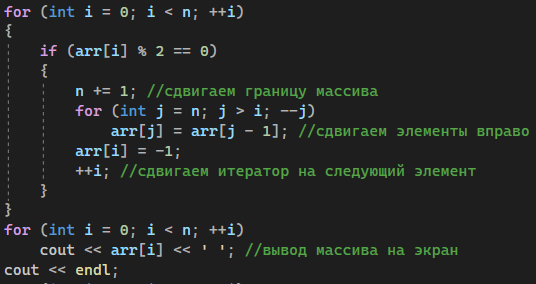
Если предыдущий элемент поседовательности >= следующему, то flag = false, и цикл прекращает работу, а пользователю выводится сообщение о том, что последовательность не упорядочена. Иначе переходим на следующую итерацию (++j).

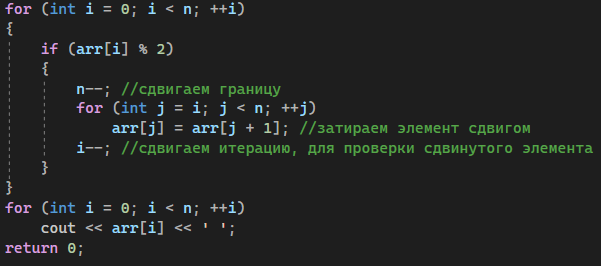
Если за всю работу цикла flag = true, то пользователю выводится надпись, что последовательность упорядочена.

1)Пример работы программы:

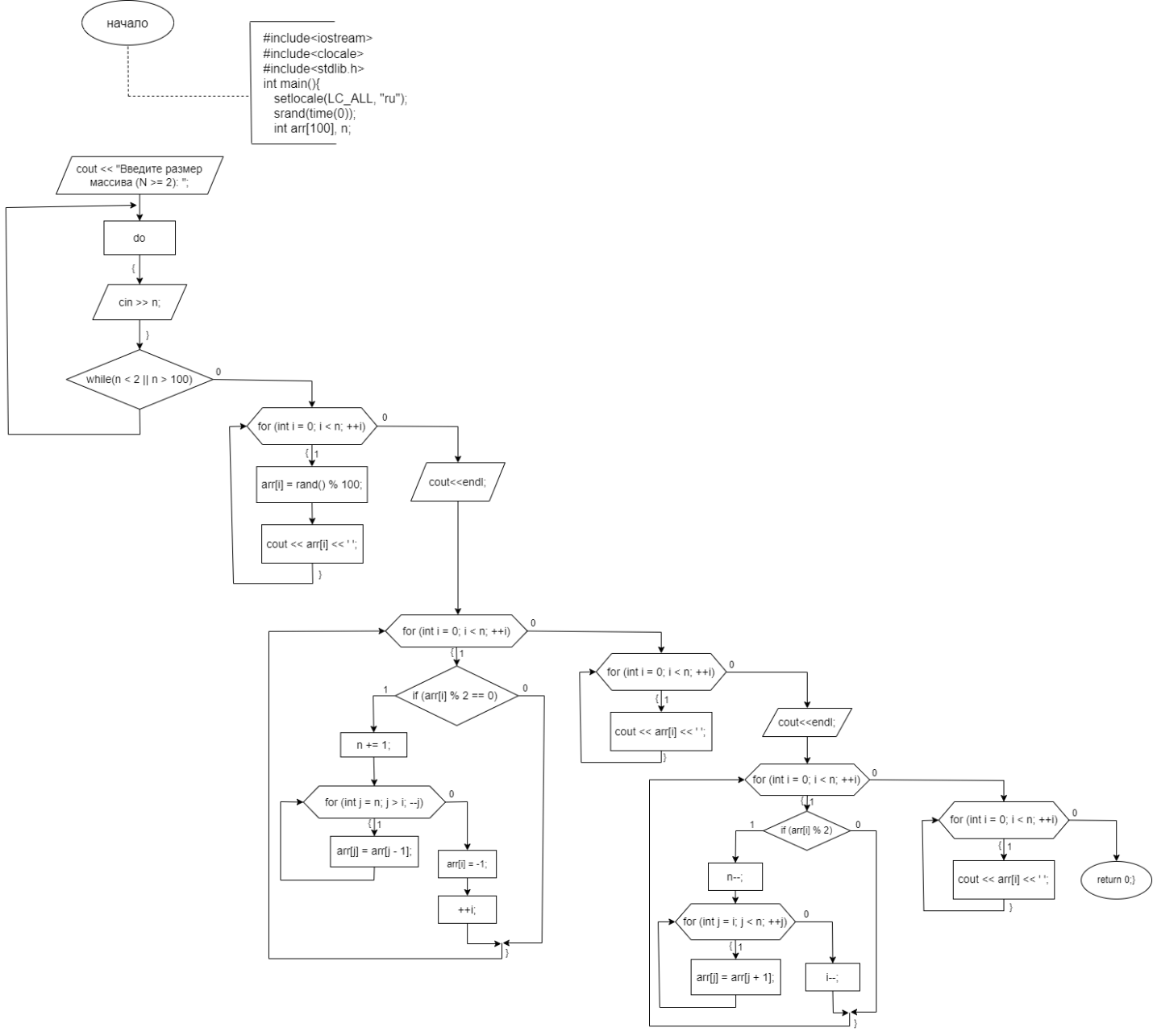




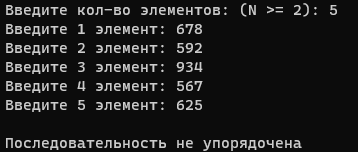


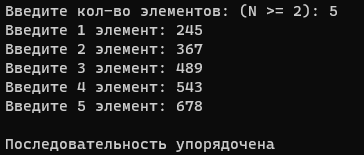


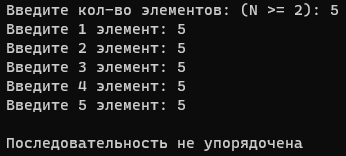
Блок-схема:

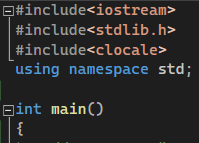


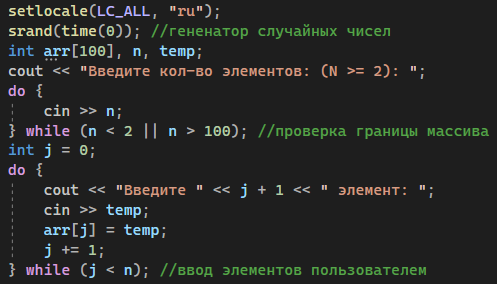
2)Пример работы программы:

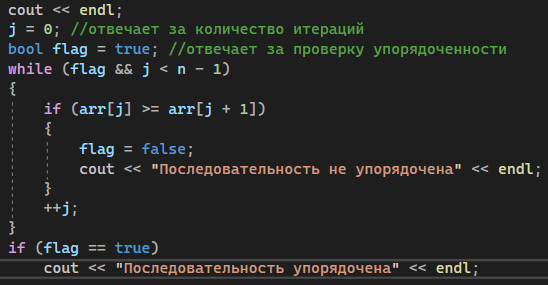












Блок-схема:

