Федеральное агентство по образованию

Государственное общеобразовательное учреждение высшего профессионального образования

**Пермский государственный технический университет**

**Лабораторная работа:**

Одномерные массивы

Выполнил:

студент группы РИС-23-2б

Колосов Данил Евгеньевич

Проверила:

доцент кафедры ИТАС

Ольга Андреевна Полякова

2024 г.

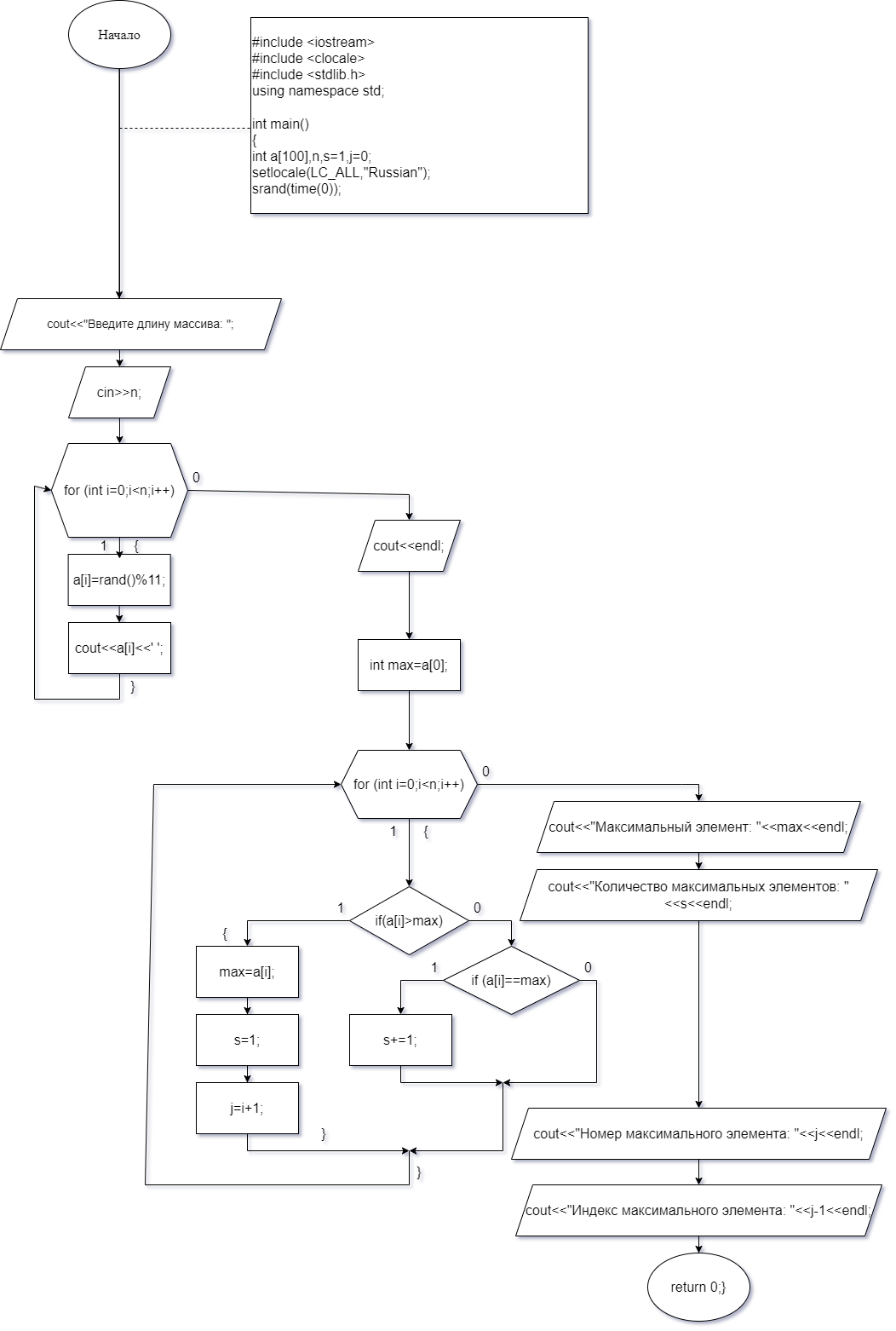
1. Максимальные числа

Дано: массив a целых чисел

Найти: максимальный элемент, его индекс и номер, если максимальных элементов несколько, то посчитать их количество.

Построение алгоритма

1. За максимальный элемент примем первый элемент массива.
2. Дальше будем сравнивать его со всеми последующими элементами и если найдётся элемент больше максимального, то в переменную, отвечающую за хранения максимального элемента, будет записан найденный.
3. При наличии нескольких максимальных, будет происходить их подсчёт в переменной s, которая, при обнаружении нового максимального элемента будет приравниваться 1.

Блок схема

Код программы

#include <iostream>

#include <clocale>

#include <stdlib.h>

using namespace std;

int main()

{

int a[100],n,s=1,j=0;

setlocale(LC\_ALL,"Russian");

srand(time(0));

cout<<"Введите длину массива: ";

cin>>n;

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

{a[i]=rand()%11;

cout<<a[i]<<' ';}

cout<<endl;

int max=a[0];

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

if(a[i]>max)

{max=a[i];

s=1;

j=i+1;

}

else

{if (a[i]==max)

s+=1;}}

cout<<"Максимальный элемент: "<<max<<endl;

cout<<"Количество максимальных элементов: "<<s<<endl;

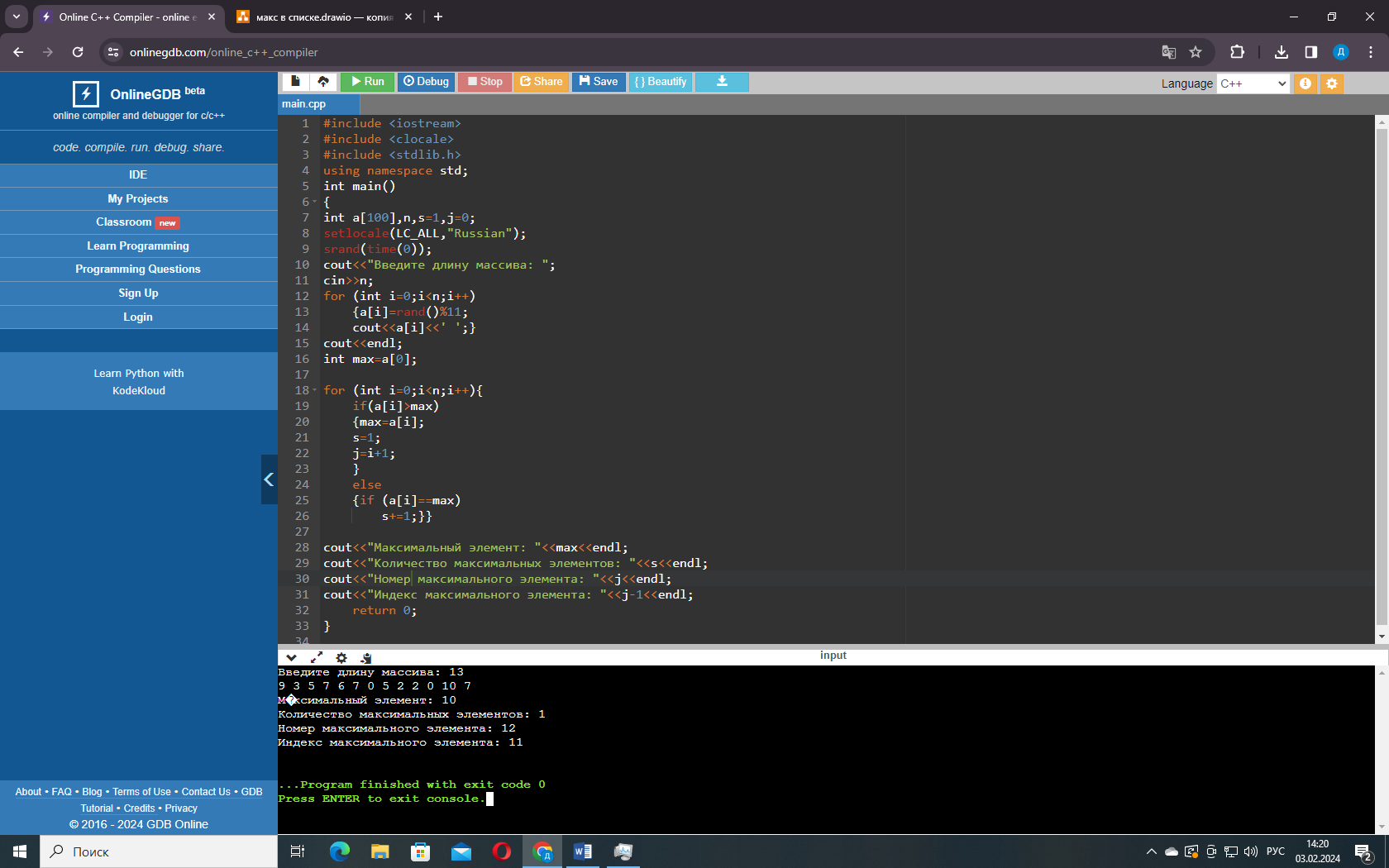
cout<<"Номер максимального элемента: "<<j<<endl;

cout<<"Индекс максимального элемента: "<<j-1<<endl;

return 0;

}

Результаты работы



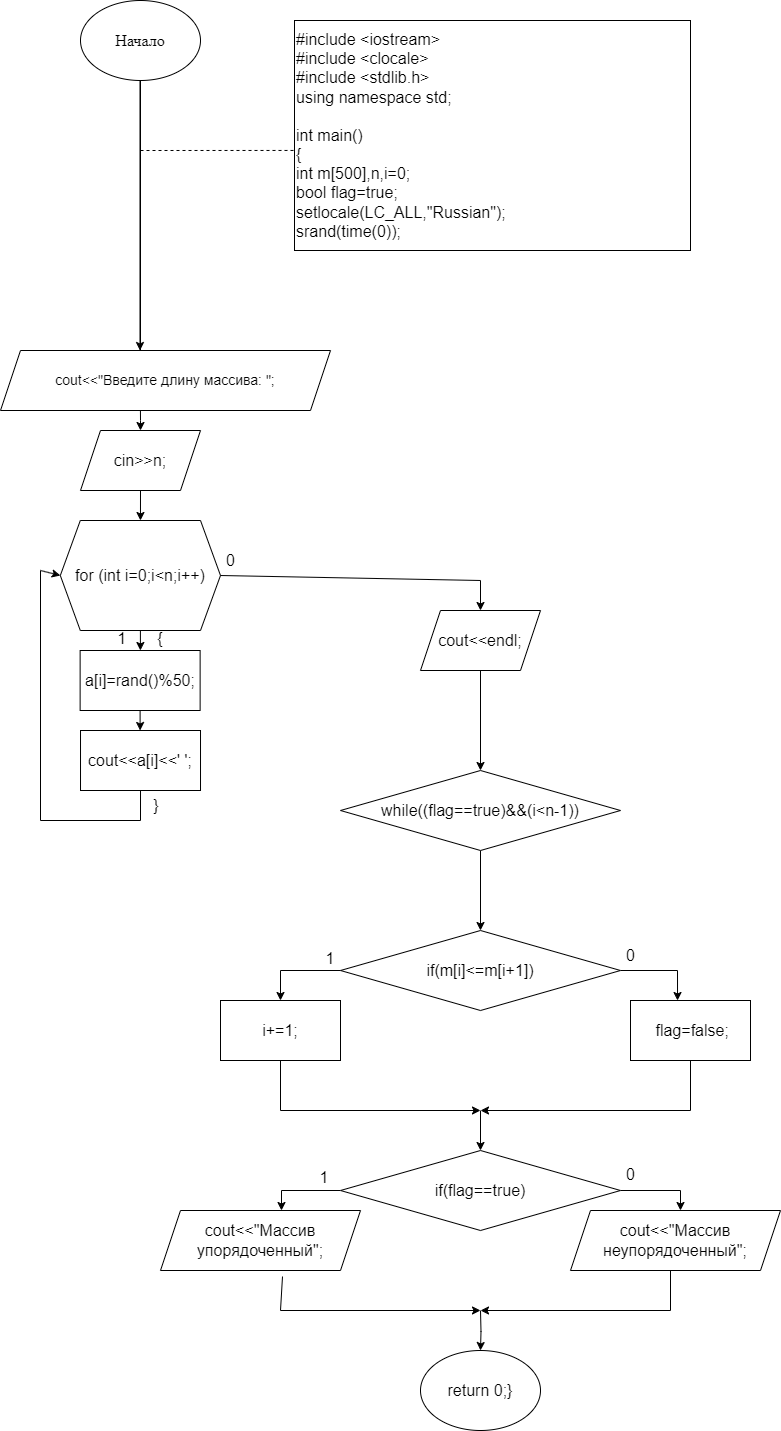
1. Упорядоченный массив

Дано: массив m целых чисел

Выяснить: является ли массив упорядоченным по возрастанию.

Построение алгоритма

1. Цикл будет поисковый, так как две причины окончания: выяснили, что массив не упорядоченный или закончился массив.
2. Задача с флажком, который меняет значение, если оказывается, что последующий элемент меньше предыдущего.

Блок схема

Код программы

#include <iostream>

#include <clocale>

#include <stdlib.h>

using namespace std;

int main()

{

int m[500],n,i=0;

bool flag=true;

setlocale(LC\_ALL,"Russian");

srand(time(0));

cout<<"Введите длину массива: ";

cin>>n;

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

{m[i]=rand()%50;

cout<<m[i]<<' ';}

cout<<endl;

while((flag==true)&&(i<n-1)){

if(m[i]<=m[i+1])

i+=1;

else

flag=false;}

if(flag==true)

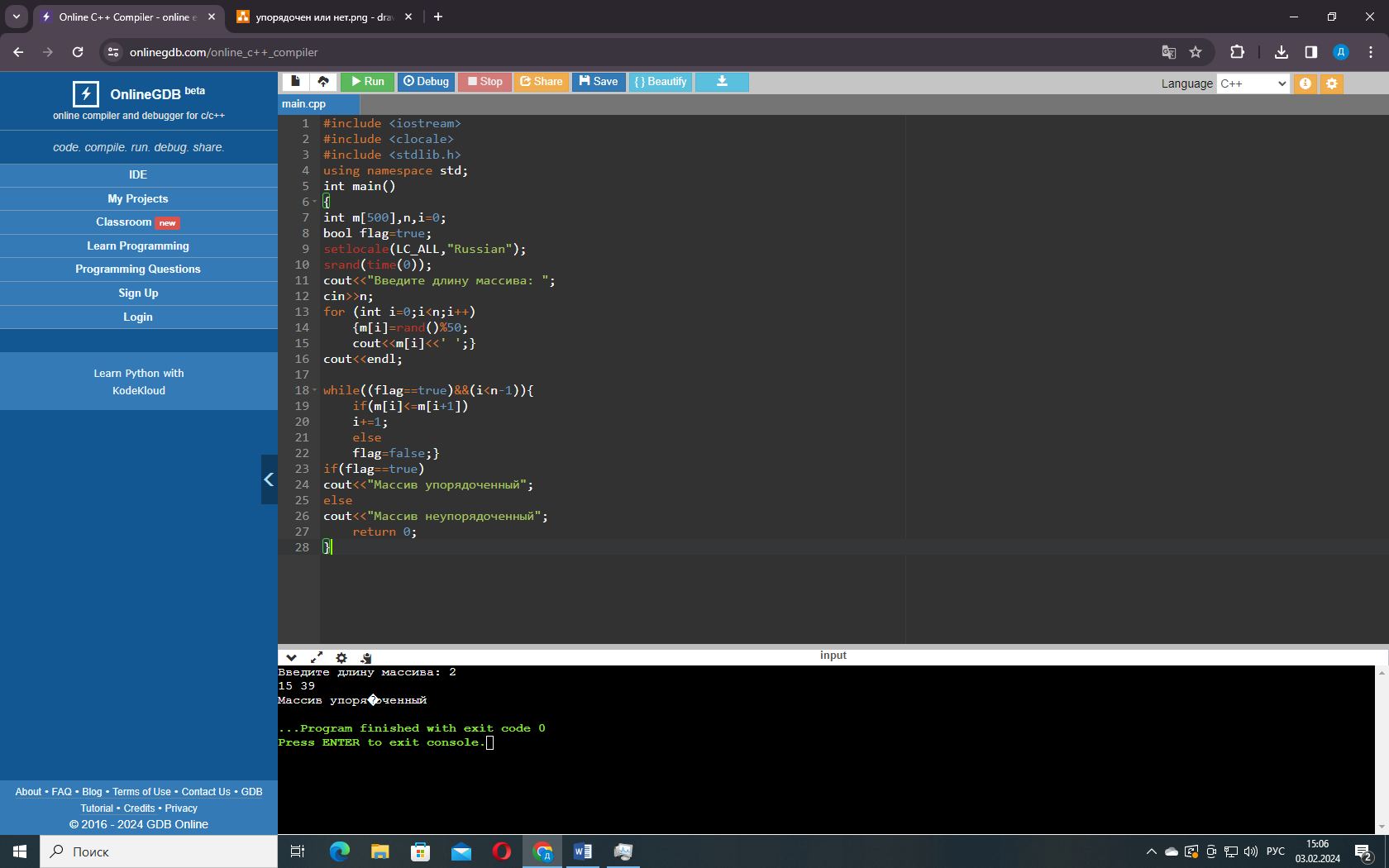
cout<<"Массив упорядоченный";

else

cout<<"Массив неупорядоченный";

return 0;

}

Результаты работы

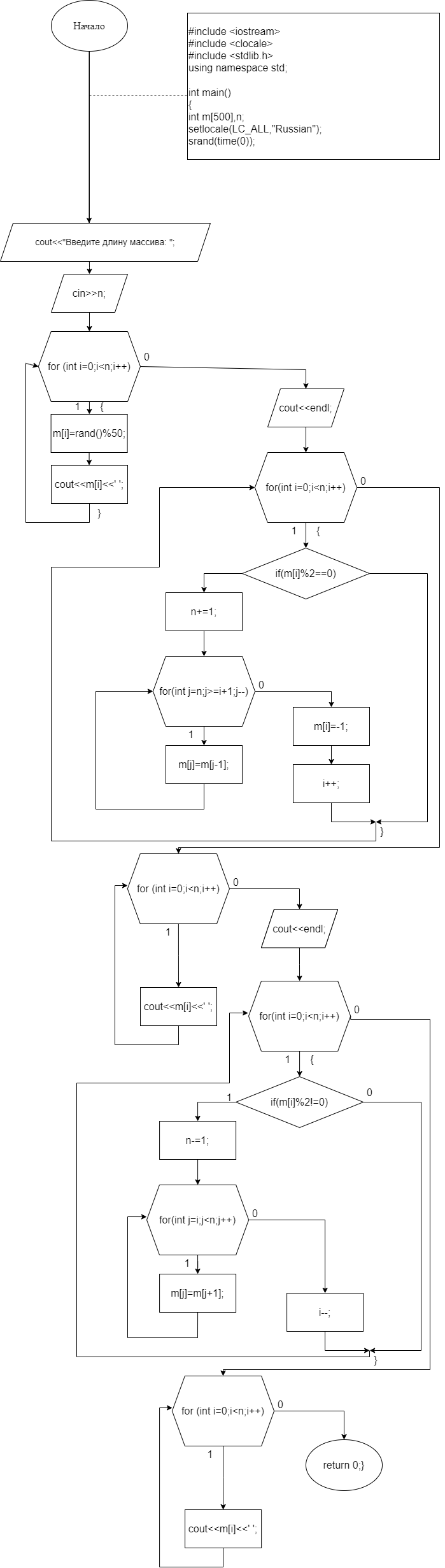
1. Чётные и нечётные элементы

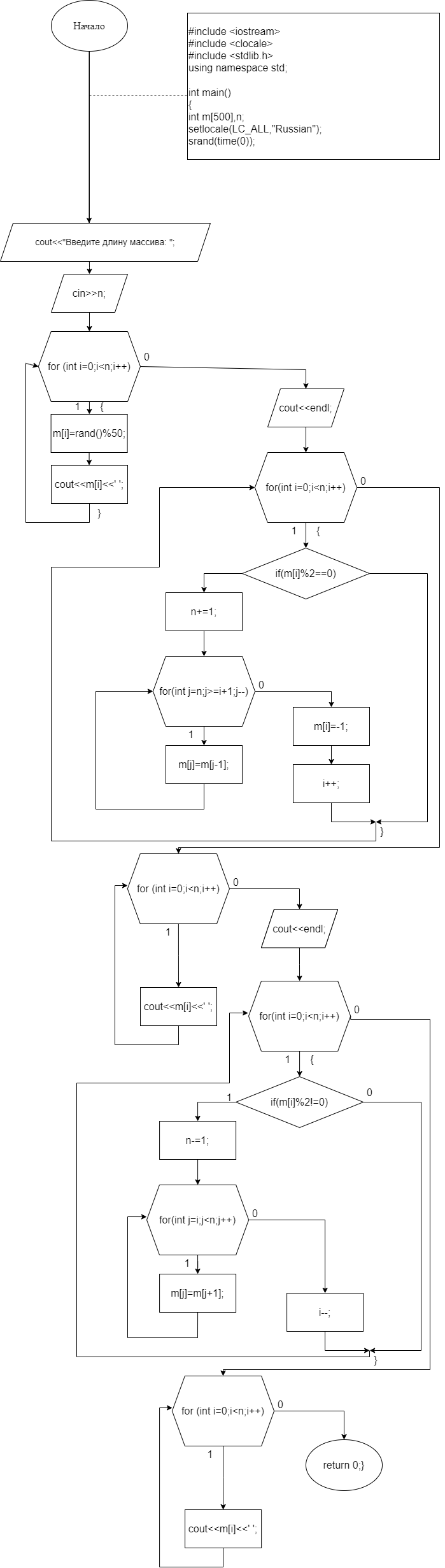
Дано: массив m целых чисел

Выполнить: перед каждым чётным элементом массива поставить -1; из полученного массива удалить все нечётные элементы.

Построение алгоритма

1. Будут использоваться вложенные циклы.
2. Внешний цикл отвечает за перебор элементов массива и проверку их на чётность.
3. Внутренний будет осуществлять сдвиг необходимых элементов в право, чтобы была возможность поставить -1 перед нужным элементом, не стирая стоящий перед ним.
4. Удаление нечётных элементов будет также осуществлено вложенными циклами, аналогично предыдущим пунктам, но элементы будут сдвигаться в лево, на место удалённого элемента.

Блок схема

Код программы

#include <iostream>

#include <clocale>

#include <stdlib.h>

using namespace std;

int main()

{

int m[500],n;

setlocale(LC\_ALL,"Russian");

srand(time(0));

cout<<"Введите длину массива: ";

cin>>n;

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

{m[i]=rand()%50;

cout<<m[i]<<' ';}

cout<<endl;

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

if(m[i]%2==0)

{n+=1;

for(int j=n;j>=i+1;j--)

{m[j]=m[j-1];}

m[i]=-1;

i++;

}

}

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

cout<<m[i]<<' ';

cout<<endl;

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

if(m[i]%2!=0){

n--;

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

m[j]=m[j+1];

}

i--;

}

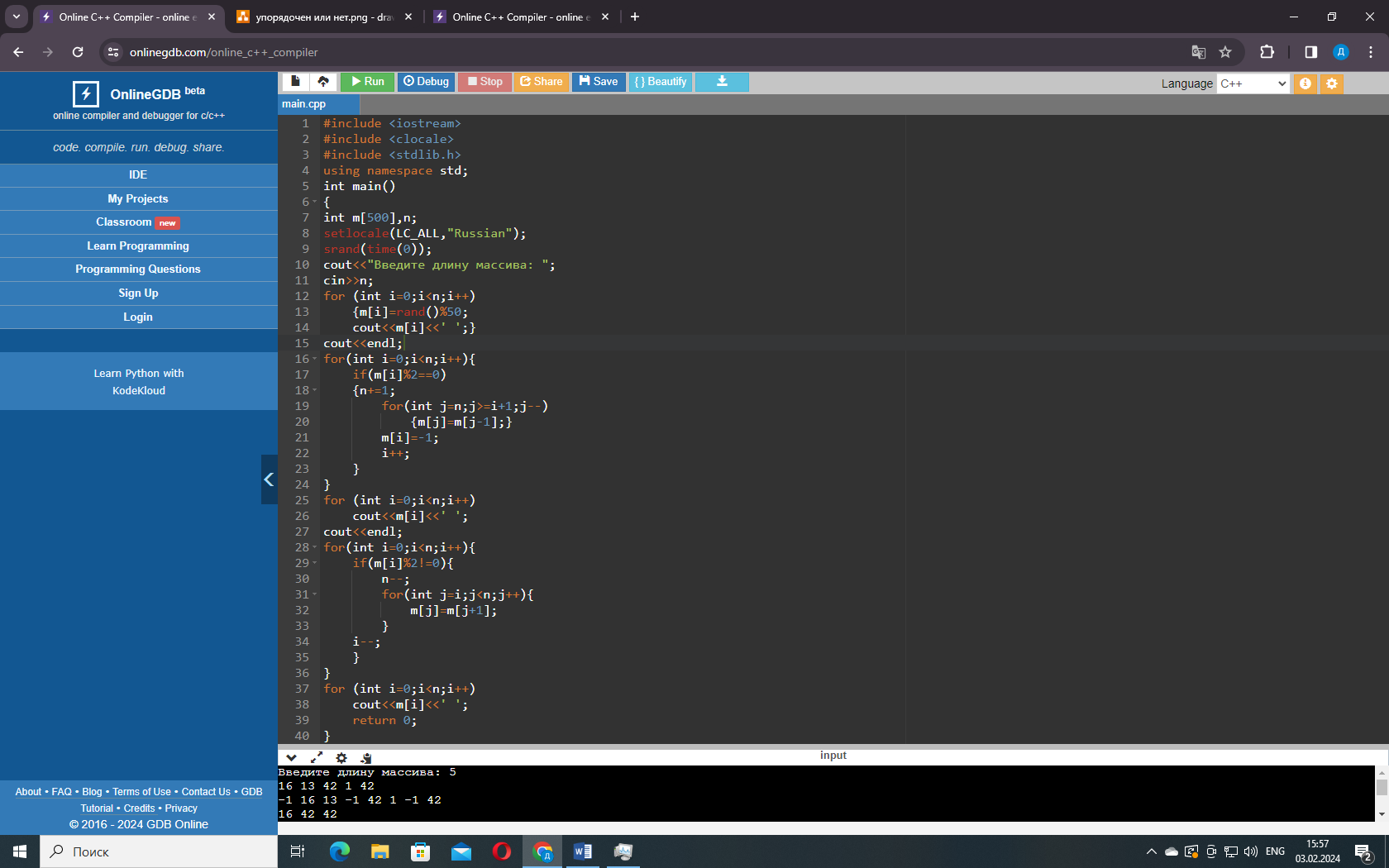
}

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

cout<<m[i]<<' ';

return 0;

}

Результаты работы

1. Кольцо из массива (вариант 18)

Дано: массив m целых чисел

Выполнить:

1) Реализовать с использованием массива однонаправленное

кольцо (просмотр возможен слева направо, от последнего

элемента можно перейти к первому).

2) Распечатать полученный массив, начиная с К-ого элемента

и до К-1.

3) Добавить в кольцо первый и последний элементы.

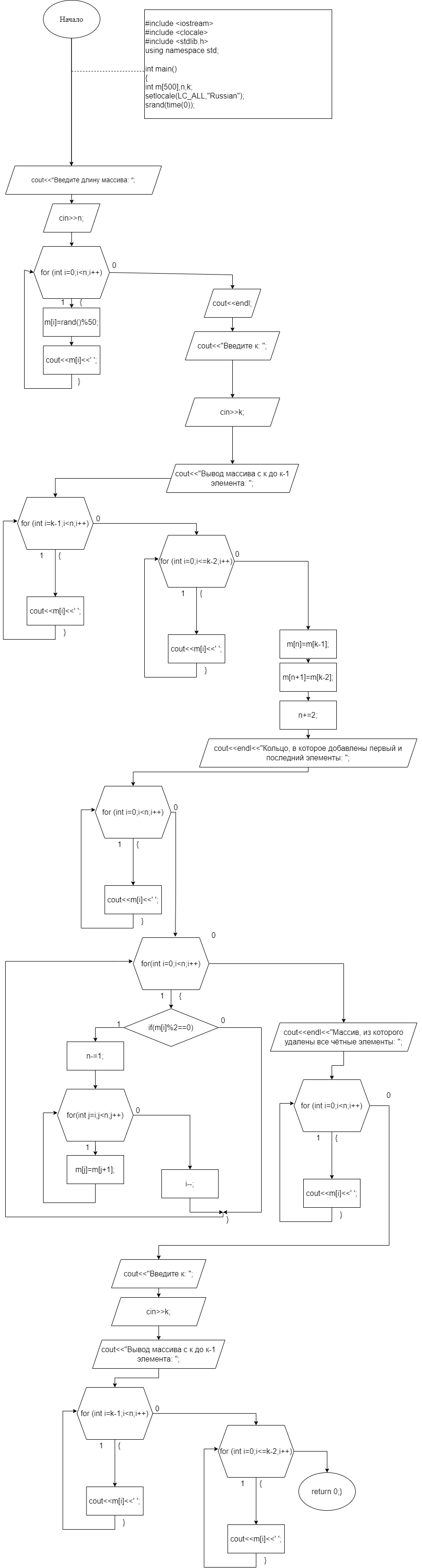
4) Удалить из кольца четные элементы.

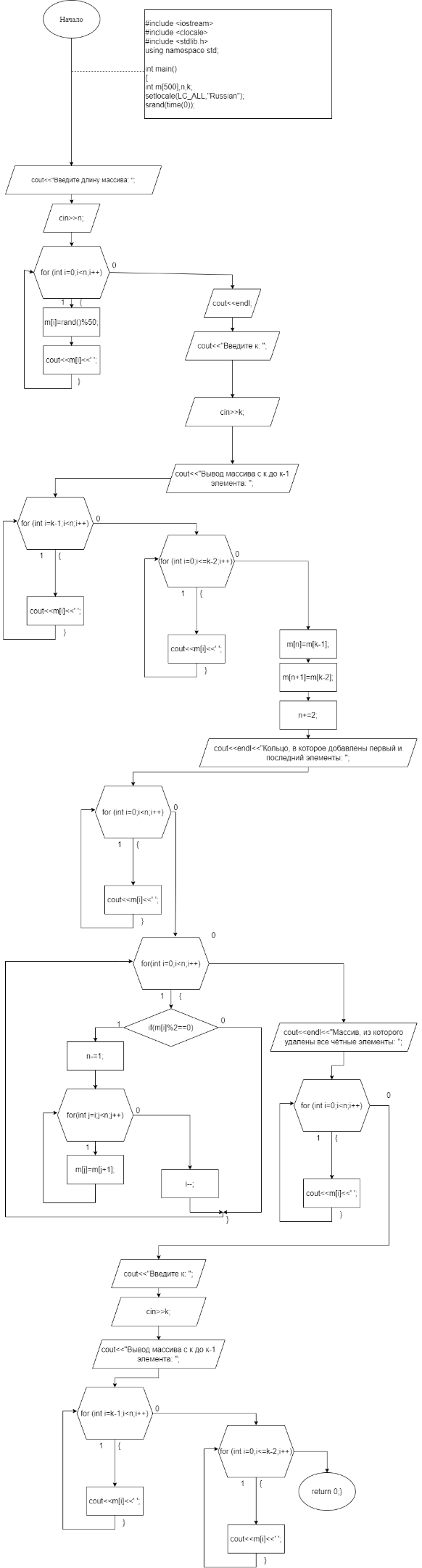
5) Распечатать полученный массив, начиная с К-ого элемента

и до К-1.

Построение алгоритма

1. Вывод кольца будет осуществлён двумя циклами, один из которых будет выводить с к-го до последнего элемента массива, а другой с первого до (к-1)-го.
2. Внешний цикл отвечает за перебор элементов массива и проверку их на чётность.
3. Внутренний будет осуществлять сдвиг необходимых элементов в право, чтобы была возможность поставить -1 перед нужным элементом, не стирая стоящий перед ним.
4. Удаление чётных элементов будет осуществлено вложенными циклами.
5. Внешний цикл отвечает за перебор элементов массива и проверку их на чётность.
6. Внутренний будет осуществлять сдвиг необходимых элементов в лево, на место чётного, тем самым «затирая» его.

Блок схема



Код программы

#include <iostream>

#include <clocale>

#include <stdlib.h>

using namespace std;

int main()

{

int m[500],n,k;

setlocale(LC\_ALL,"Russian");

srand(time(0));

cout<<"Введите длину массива: ";

cin>>n;

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

{m[i]=rand()%50;

cout<<m[i]<<' ';}

cout<<endl;

cout<<"Введите к: ";

cin>>k;

cout<<"Вывод массива с к до к-1 элемента: ";

for (int i=k-1;i<n;i++){

cout<<m[i]<<' ';

}

for (int i=0;i<=k-2;i++)

cout<<m[i]<<' ';

m[n]=m[k-1];

m[n+1]=m[k-2];

n+=2;

cout<<endl<<"Кольцо, в которое добавлены первый и последний элементы: ";

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

cout<<m[i]<<' ';

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

if(m[i]%2==0){

n--;

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

m[j]=m[j+1];}

i--;

}

}

cout<<endl<<"Массив, из которого удалены все чётные элементы: ";

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

cout<<m[i]<<' ';

cout<<endl<<"Введите к: ";

cin>>k;

cout<<"Вывод массива с к до к-1 элемента: ";

for (int i=k-1;i<n;i++){

cout<<m[i]<<' ';

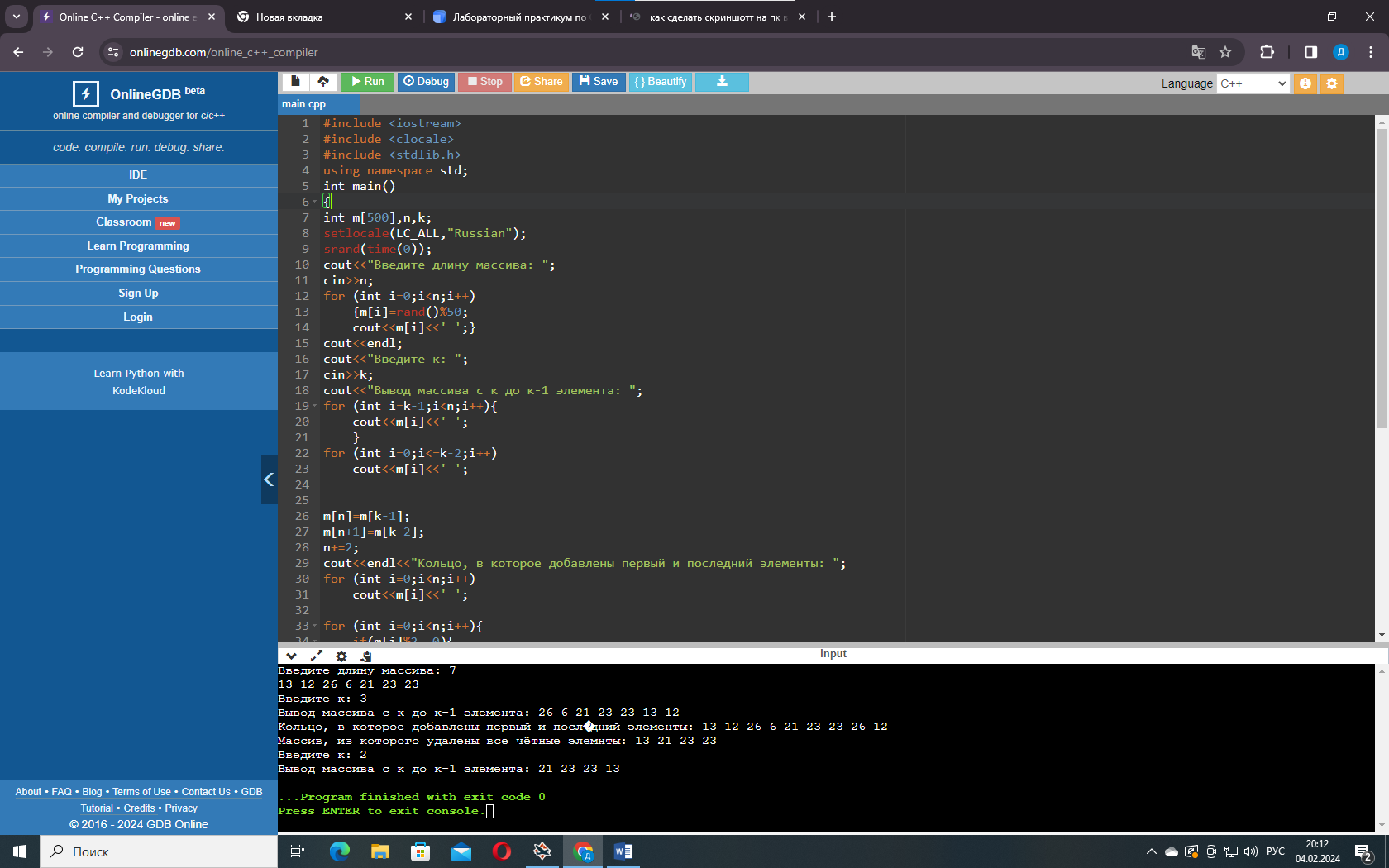
}

for (int i=0;i<=k-2;i++)

cout<<m[i]<<' ';

return 0;

}

Результаты работы

Ссылка на гит