Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования

**Пермский национальный исследовательский политехнический университет**

Электротехнический факультет

Кафедра информационных технологий и автоматизированных систем

**ОТЧЕТ**

**Тема:** Одномерные массивы

Семестр: 2

Выполнил студент РИС-23-2б:

Серебряков Егор Константинович

(дата, подпись)

Проверила:

О. А. Полякова

(дата, подпись)

Пермь 2024

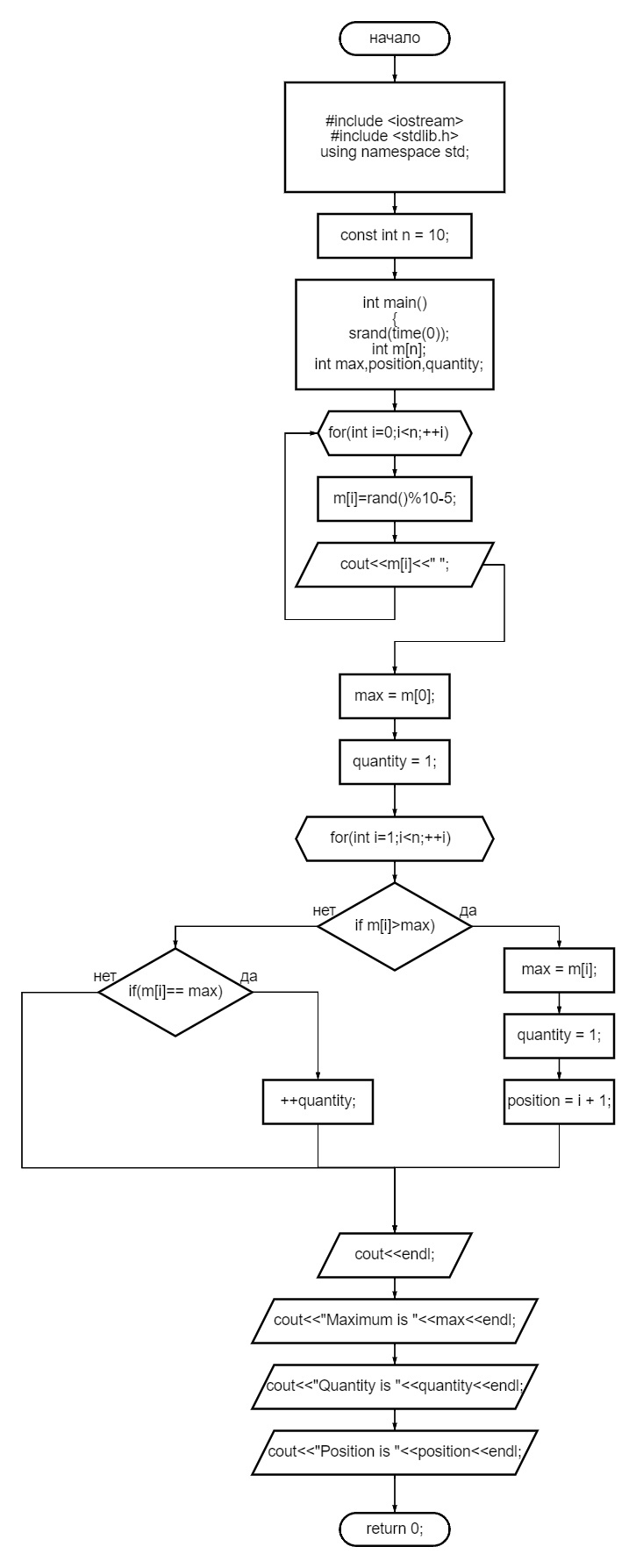
**Лабораторная работа №1**

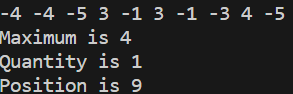


Задача: построить алгоритм, который выведет максимальный элемент одномерного массива, его номер и кол-во повторений, если таких элементов несколько.

Шаги решения:

1. Создаём массив с указанным пользователем значением длины, ячейки которого заполнены случайными числами
2. Проходимся по массиву, используя цикл for. Находим максимальный элемент и его основные данные
3. В следующих итерациях цикла в зависимости от значения проверяемого элемента, программа должна будет либо прибавить к счетчику повторяющихся максимальных чисел +1, либо обнулить значение счетчика и записать новое значение и другие данные, связанные с найденным максимальным числом
4. Вывести требуемый результат
5. #include <iostream>
6. #include <stdlib.h>
7. using namespace std;
8. const int n = 10;
9. int main()
10. {
11. srand(time(0));
12. int m[n];
13. int max,position,quantity;
14. for(int i=0;i<n;++i)        //Заполнение массива случайными числами
15. {
16. m[i] = rand()%10-5;
17. cout<<m[i]<<" ";
18. }
19. max = m[0];
20. quantity = 1;
21. for(int i=1;i<n;++i)        //Цикл для перебора массива
22. {
23. if(m[i]>max)            //Если нашлось новое максимальное число, то обнуляем счётчики, записываем число в max
24. {
25. max = m[i];
26. quantity = 1;
27. position = i + 1;
28. }
29. else
30. {
31. if(m[i]== max)      //Если находим число равное максимальному, то прибавляем +1 к счёту равных элементов с элементом max
32. {
33. ++quantity;
34. }
35. }
36. }
37. cout<<endl;
38. cout<<"Maximum is "<<max<<endl;         //Вывод максимального значения
39. cout<<"Quantity is "<<quantity<<endl;   //Вывод кол-ва повторяющегося макс.элемента
40. cout<<"Position is "<<position<<endl;   //Вывод позиции макс.элемента
41. return 0;
42. }





**Лабораторная работа №2**

Задача: создать массив, проверить упорядочен ли он по возрастанию

Шаги решения:

1. Создаём массив с помощью ручного ввода, указываем его длину и пишем значения для каждого элемента
2. Используем цикл for с булевой переменной, изменяем значение флажка, если значения сравниваемых элементов не будут удовлетворять условию и выводим сообщение о том, что массив не упорядочен
3. Проверяем с помощью условия значение флажка, если истина, пишем, что массив упорядочен

#include <iostream>

#include <stdlib.h>

using namespace std;

const int n = 10;

int main()

{

    srand(time(0));

    int m[n];

    bool flag = true;

    for(int i=0;i<n;++i)        //Ручной ввод элементов массива, всего их 10, можно реализовать ввод длины

    {

        cout<<"Enter the "<<i+1<<" element:";

        cin>>m[i];

        cout<<endl;

    }

    cout<<"This is your array:";        //Вывод массива с введенными числами, если он не упорядочен, программа выведет это

    for(int i=0;i<n-1;++i)

    {

        cout<<m[i]<<" ";

        if(m[i]<=m[i+1])

        {

            //Do nothing

        }

        else

        {

            flag = false;

        }

    }

    if(flag == true)        //Вывод сообщения о том, что массив упорядочен

    {

        cout<<endl<<"Array is ordered";

    }

    else

    {

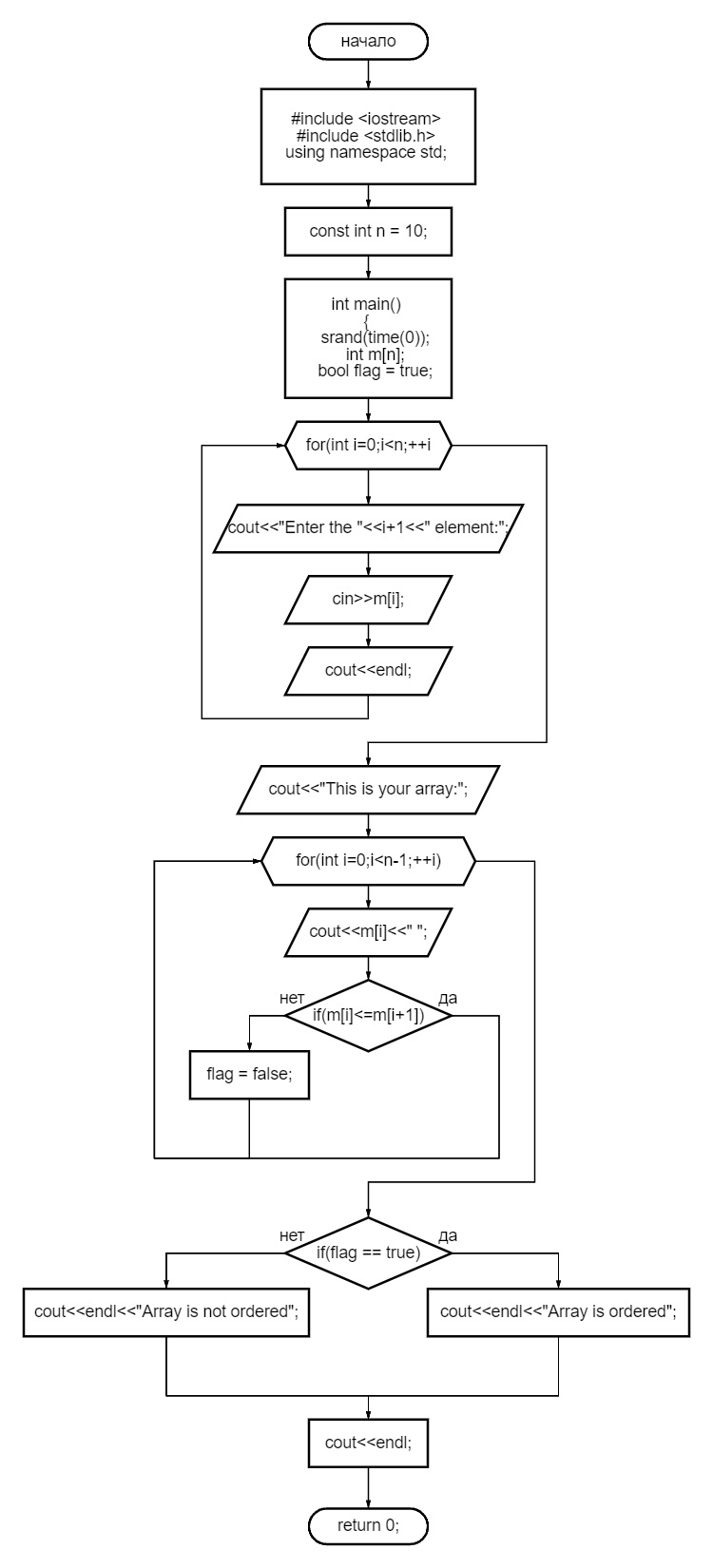
        cout<<endl<<"Array is not ordered";

    }

    cout<<endl;

    return 0;

}





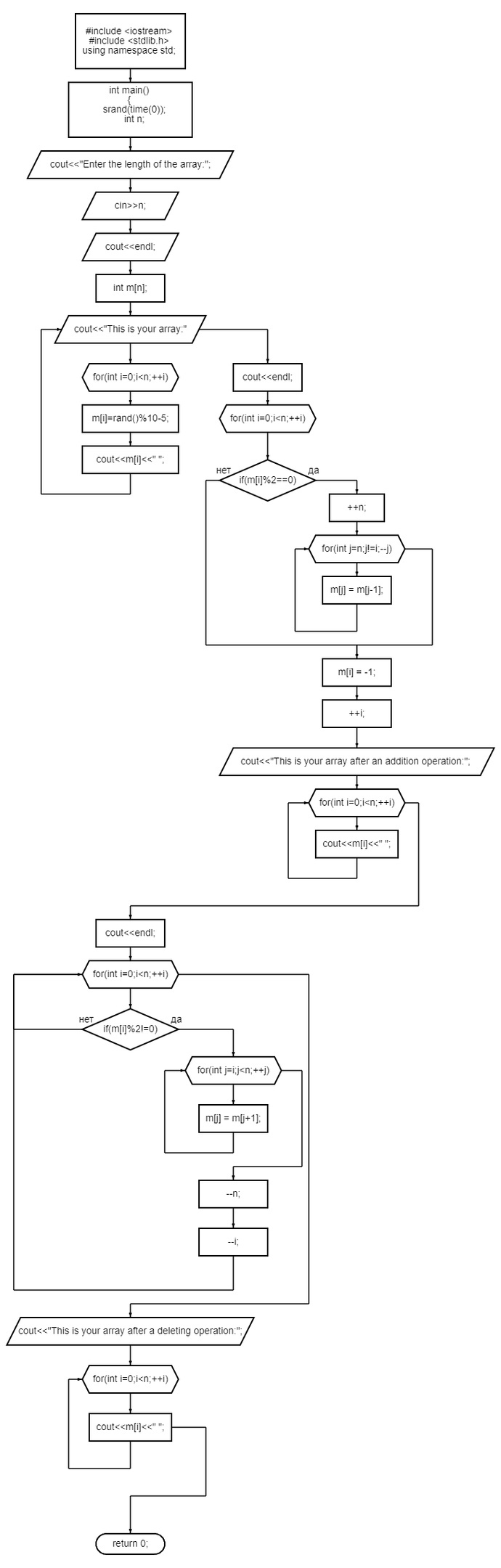


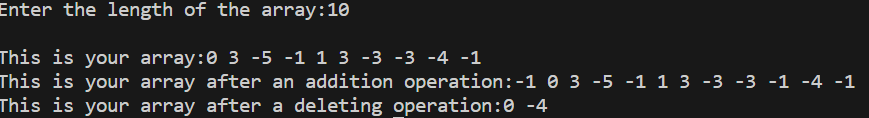
**Лабораторная работа №3**

Задача: создать массив, проставить перед четными числами -1, вывести массив, удалить все нечетные элементы, вывести массив

Шаги решения:

1. Создаём массив случайных чисел
2. Создаём внешний цикл for для перебора элементов, внутренний цикл for будет осуществлять сдвиг элементов вправо на 1 ячейку, чтобы вставить -1, не затирая минус единицей число. Значение длины массива изменяем на +1, если добавляем -1 перед четным числом
3. Выводим массив
4. С помощью внешнего цикла for перебираем элементы, находим нечётные с помощью условия и удаляем элемент, сдвигая часть массива на 1 влево, используя внутренний цикл for
5. Выводим массив
6. #include <iostream>
7. #include <stdlib.h>
8. using namespace std;
9. int main()
10. {
11. srand(time(0));
12. int n;
13. cout<<"Enter the length of the array:";     //Ввод длины будущего массива
14. cin>>n;
15. cout<<endl;
16. int m[n];
17. cout<<"This is your array:";        //Создание массива случайных чисел
18. for(int i=0;i<n;++i)
19. {
20. m[i]=rand()%10-5;
21. cout<<m[i]<<" ";
22. }
23. cout<<endl;
24. for(int i=0;i<n;++i)        //Часть кода, отвечающая за проставление -1 перед четным числом
25. {
26. if(m[i]%2==0)
27. {
28. ++n;
29. for(int j=n;j!=i;--j)
30. {
31. m[j] = m[j-1];
32. }
33. m[i] = -1;
34. ++i;
35. }
37. }
38. cout<<"This is your array after an addition operation:";        //Вывод массива с -1 перед четными числами
39. for(int i=0;i<n;++i)
40. {
41. cout<<m[i]<<" ";
42. }
43. cout<<endl;
44. for(int i=0;i<n;++i)        //Часть кода, отвечающая за удаление нечетных элементов массива
45. {
46. if(m[i]%2!=0)
47. {
48. for(int j=i;j<n;++j)
49. {
50. m[j] = m[j+1];
51. }
52. --n;
53. --i;
54. }
55. }
56. cout<<"This is your array after a deleting operation:";         //Вывод массива после удаления из него нечётных элементов
57. for(int i=0;i<n;++i)
58. {
59. cout<<m[i]<<" ";
60. }
61. return 0;
62. }





**Лабораторная работа №4**

Задача:

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) Упорядочить элементы по возрастанию

4) Удалить из кольца четные элементы.

5) Распечатать полученный массив, начиная с К-ого элемента и до К-1

Шаги решения:

1. Создаём массив случайных чисел
2. Вводим шаг для кольца
3. Используем два внешних цикла for для вывода массива со сдвигом k
4. Используем метод пузырька для упорядочивания массива по возрастанию
5. Выводим упорядоченный массив
6. Удаляем из массива четные элементы с помощью перебора чисел
7. Вводим шаг для кольца
8. Используем два внешних цикла for для вывода массива со сдвигом k, если длина массива больше одного, иначе программа выведет, что длина массива меньше или равна 1
9. #include <iostream>
10. #include <stdlib.h>
11. using namespace std;
12. int main()
13. {
14. srand(time(0));
15. int n;
16. cout<<"Enter the lenght of the array:"; //Ввод длины будущего массива
17. cin>>n;
18. int m[n];
19. cout<<endl;
20. cout<<"This is your array:";    //Создание массива случайных чисел
21. for(int i=0;i<n;++i)
22. {
23. m[i] = rand()%10-5;
24. cout<<m[i]<<" ";
25. }
26. cout<<endl;
27. int k;      //Ввод шага для кольца
28. do
29. {
30. cout<<"Enter the step for the circle:";
31. cin>>k;
32. cout<<endl;
33. }
34. while(k>n);
35. cout<<"Array with entered step:";       //Вывод кольца с шагом k
36. for(int i=k-1;i<n;++i)
37. {
38. cout<<m[i]<<" ";
39. }
40. for(int i=0;i<k-1;++i)
41. {
42. cout<<m[i]<<" ";
43. }
44. cout<<endl;
45. int third\_cup;      //Сортировка массива методом пузырька
46. for(int i=n-1;i!=0;--i)
47. {
48. for(int j=0;j<i;++j)
49. {
50. if(m[j]>m[j+1])
51. {
52. third\_cup = m[j];
53. m[j] = m[j+1];
54. m[j+1] = third\_cup;
55. }
56. }
57. }
58. cout<<"Array with sort:";       //Вывод отсортированного массива
59. for(int i=0;i<n;++i)
60. {
61. cout<<m[i]<<" ";
62. }
63. cout<<endl;
64. for(int i=0;i<n;++i)        //Удаление четных элементов
65. {
66. if(m[i]%2==0)
67. {
68. for(int j=i;j<n;++j)
69. {
70. m[j] = m[j+1];
71. }
72. --n;
73. --i;
74. }
75. }
76. cout<<"Array after a deleting operation:";      //Вывод массива после удаление четных элементов
77. for(int i=0;i<n;++i)
78. {
79. cout<<m[i]<<" ";
80. }
81. cout<<endl;
83. if(n<=1)        //Вывод результата, если длина массива равна или меньше нулю/запрос на шаг сдвига массива и вывод массива со сдвигом
84. {
85. cout<<"Array's length <= 1";
86. }
87. else
88. {
89. do                              //Ввод шага для кольца
90. {
91. cout<<"Enter the step for the circle:";
92. cin>>k;
93. cout<<endl;
94. }
95. while(k>n);
96. cout<<"Array with entered step:";       //Вывод кольца с шагом k
97. for(int i=k-1;i<n;++i)
98. {
99. cout<<m[i]<<" ";
100. }
101. for(int i=0;i<k-1;++i)
102. {
103. cout<<m[i]<<" ";
104. }
105. }
106. return 0;
107. }

