

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1
дисциплины «Алгоритмизация»
Вариант ____

Выполнил:
Дудкин Константин Александрович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»
направление «Программное
обеспечение средств вычислительной
техники и автоматизированных
систем»,
очная форма обучения

(подпись)

Руководитель практики:
доцент кафедры инфокоммуникаций,
кандидат технических наук
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема:

Цель:

Порядок выполнения работы

1. Открыл сайт sdamgia.ru, выбрал пункт «Информатика» - «ЕГЭ». Затем выбрал по одному заданию 16, 17, 18 и нажал на «Составить вариант».
2. Приступил к выполнению задания 16:

1 Тип 16 № 4645    

Алгоритм вычисления значения функции $F(n)$, где n – натуральное число, задан следующими соотношениями:

$F(1) = 1$

$F(2) = 3$

$F(n) = F(n-1) * n + F(n-2) * (n - 1)$, при $n > 2$

Чему равно значение функции $F(5)$?

В ответе запишите только натуральное число.

Рисунок 1 – Условие задания 16

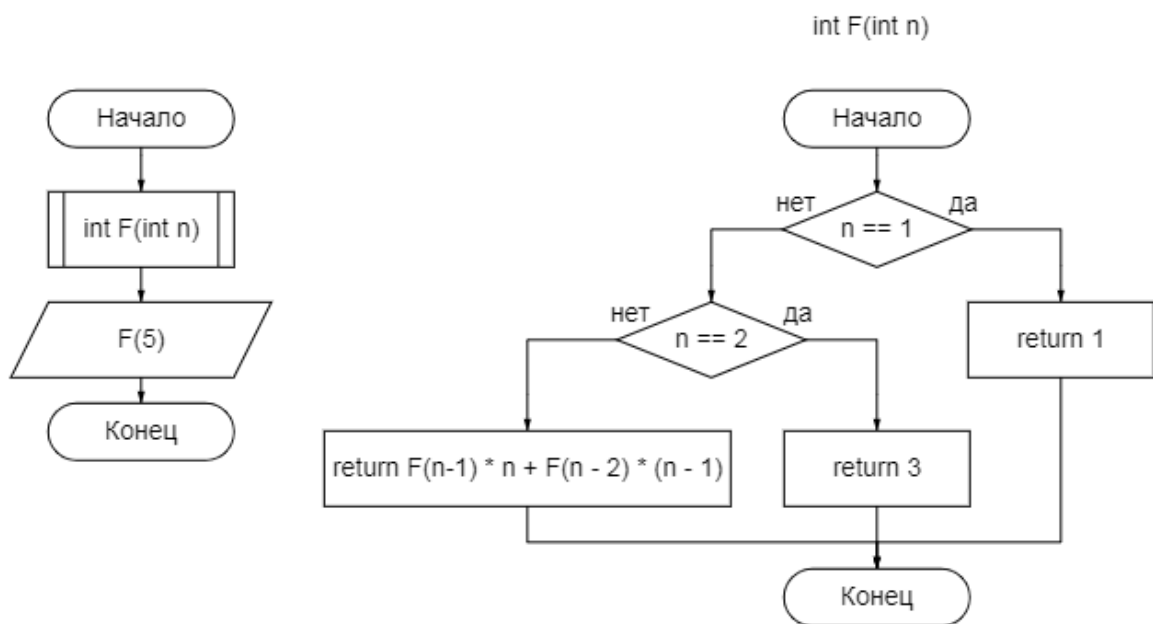
Код программы:

```
using System;

namespace Задание_16_ПР1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            int F(int n)
            {
                if (n == 1)
                {
                    return 1;
                }
                else if (n == 2)
                {
                    return 3;
                }
                else
                {
                    return F(n - 1) * n + F(n - 2) * (n - 1);
                }
            }

            int result = F(5);
            Console.WriteLine("Результат - " + result);
        }
    }
}
```

Блок-схема программы:



3. Приступил к выполнению задания 17:

43 Тип 17 № 59784

В файле содержится последовательность целых чисел. Элементы последовательности могут принимать целые значения от — 1000000 до 1000000 включительно.

Задание 17

Определите количество троек элементов в которых только одно число четырехзначное, и сумма элементов тройки меньше максимального элемента последовательности оканчивающегося на 15. В ответе запишите два числа: сначала количество найденных троек, а затем минимальную из сумм таких троек. В данной задаче под тройкой подразумевается три идущих подряд элемента последовательности.

Рисунок 2 – Условие задания 17

Код программы на C#:

```

using System;
using System.Linq;

namespace Задание_17_ПР1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Random rand = new Random();
            int[] array = new int[500];
            int max = -1000001;
            int count = 0;
            int minsum = 3000001;
            for(int i = 0; i < array.Length; i++)
            {
                array[i] = rand.Next(-1000000, 1000000);
            }
        }
    }
}

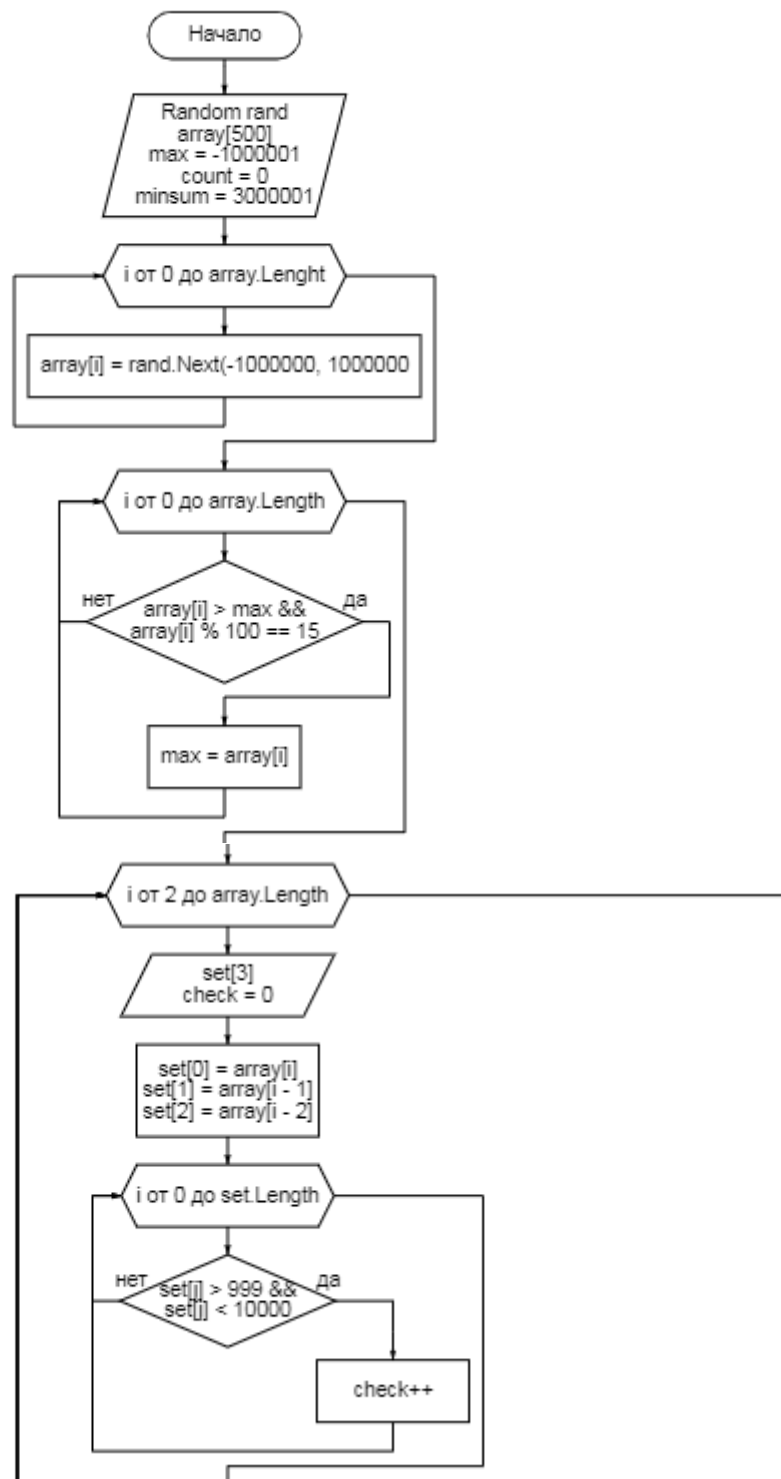
```

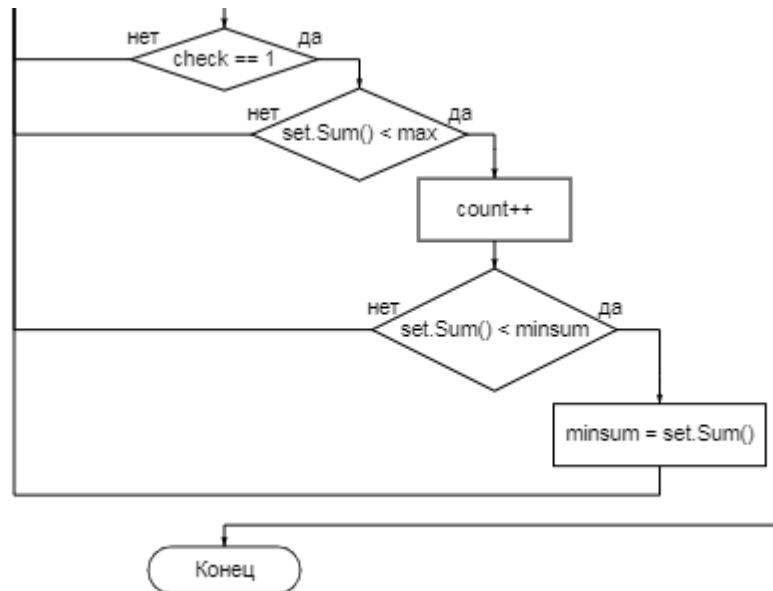
```

        for(int i = 0;i < array.Length; i++)
        {
            if (array[i] > max && array[i] % 100 == 15)
            {
                max = array[i];
            }
        }
        for (int i = 2;i < array.Length; i++)
        {
            int[] set = new int[3];
            int check = 0;
            set[0] = array[i];
            set[1] = array[i - 1];
            set[2] = array[i - 2];
            for (int j = 0; j < set.Length; j++)
            {
                if (set[j] > 999 && set[j] < 10000)
                {
                    check++;
                }
            }
            if(check == 1)
            {
                if (set.Sum() < max)
                {
                    count++;
                    if (set.Sum() < minsum)
                    {
                        minsum = set.Sum();
                    }
                }
            }
        }
        Console.WriteLine(count + ", " + minsum);
    }
}

```

Блок-схема программы:





4. Приступил к выполнению задания 18

1 Тип 18 № 27415 📁 ● ! i

Квадрат разбит на $N \times N$ клеток ($1 < N < 17$). Исполнитель Робот может перемещаться по клеткам, выполняя за одно перемещение одну из двух команд: вправо или вниз. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вниз — в соседнюю нижнюю. При попытке выхода за границу квадрата Робот разрушается. Перед каждым запуском Робота в каждой клетке квадрата лежит монета достоинством от 1 до 100. Посетив клетку, Робот забирает монету с собой; это также относится к начальной и конечной клетке маршрута Робота.

Задание 18

Откройте файл. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой верхней клетки в правую нижнюю. В ответ запишите два числа друг за другом без разделительных знаков — сначала максимальную сумму, затем минимальную.

Исходные данные представляют собой электронную таблицу размером $N \times N$, каждая ячейка которой соответствует клетке квадрата.

Пример входных данных:

1	8	8	4
10	1	1	3
1	3	12	2
2	3	5	6

Для указанных входных данных ответом должна быть пара чисел 41 и 22.

Рисунок 3 – Условие задания 18

Код программы на C#:

```

using System;

namespace Задание_18_ПР1
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Random rand = new Random();
            int n1 = rand.Next(2, 16);
            int n2 = rand.Next(2, 16);
            int[,] matrix = new int[n1,n2];
            Console.WriteLine("Текущая матрица:");
        }
    }
}

```

```

Console.WriteLine();
for (int i = 0; i < n1; i++)
{
    for (int j = 0; j < n2; j++)
    {
        matrix[i, j] = rand.Next(1, 100);
        Console.Write(matrix[i, j] + " ");
    }
    Console.WriteLine();
}

int max = 0;
int min = 0;
for(int i = 0; i < n1; i++)
{
    for(int j = 0; j < n2; j++)
    {
        if(i == 0 && j > 0)
        {
            matrix[i, j] = matrix[i, j] + matrix[i, j - 1];
        }
        else if(i > 0 && j == 0)
        {
            matrix[i, j] = matrix[i, j] + matrix[i - 1, j];
        }
        else if(i > 0 && j > 0)
        {
            if (matrix[i - 1, j] >= matrix[i, j - 1])
            {
                matrix[i, j] = matrix[i, j] + matrix[i - 1, j];
            }
            else
            {
                matrix[i, j] = matrix[i, j] + matrix[i, j - 1];
            }
        }
        if (i == n1 - 1 && j == n2 - 1)
        {
            min = matrix[i, j];
        }
    }
}

for (int i = 0; i < n1; i++)
{
    for (int j = 0; j < n2; j++)
    {
        if (i == 0 && j > 0)
        {
            matrix[i, j] = matrix[i, j] + matrix[i, j - 1];
        }
    }
}

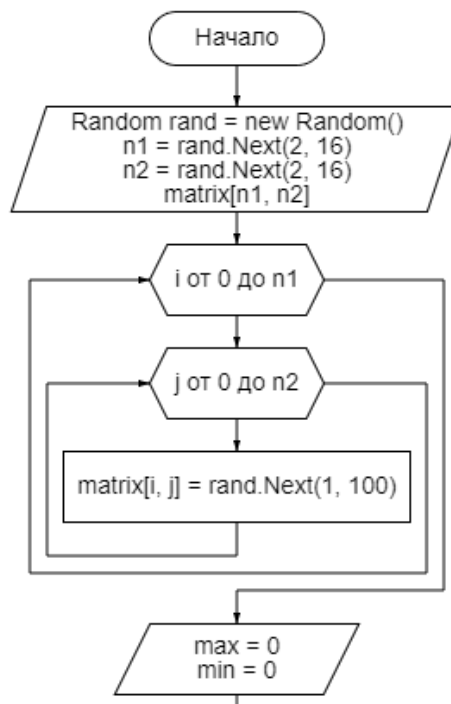
```

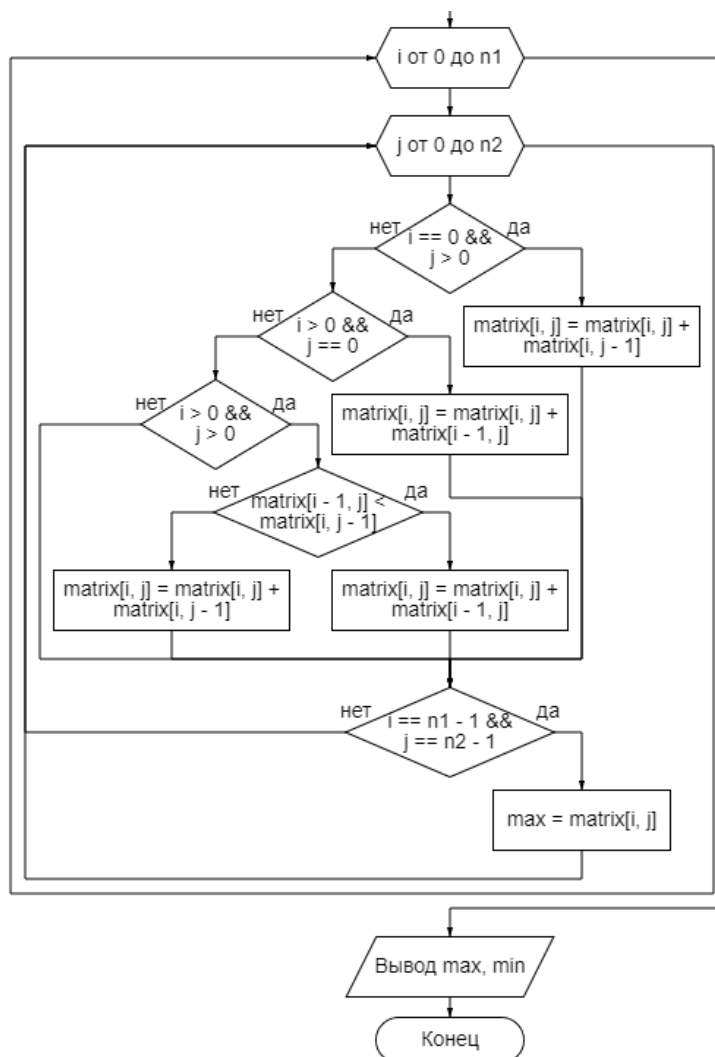
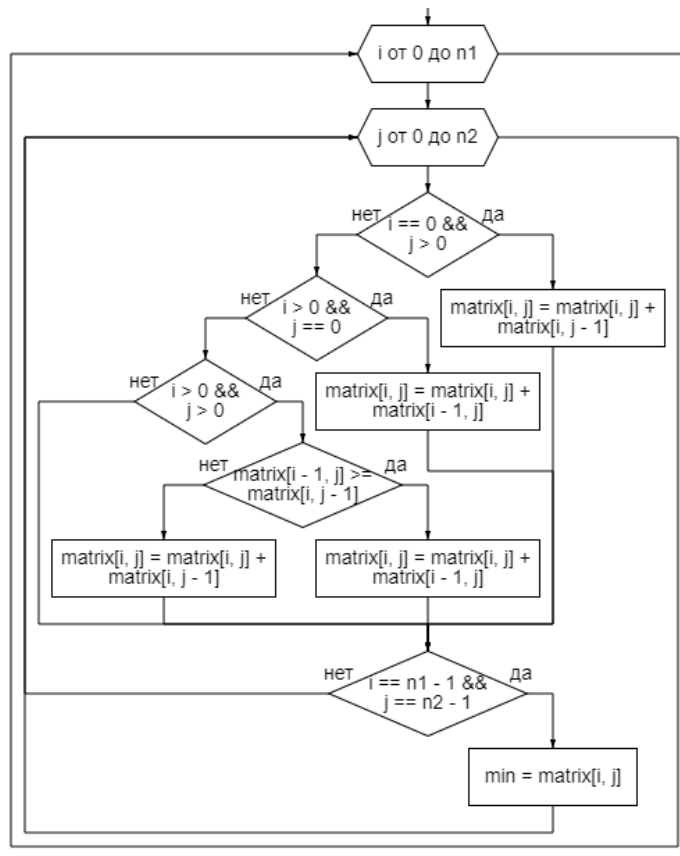
```

else if (i > 0 && j == 0)
{
    matrix[i, j] = matrix[i, j] + matrix[i - 1, j];
}
else if (i > 0 && j > 0)
{
    if (matrix[i - 1, j] < matrix[i, j - 1])
    {
        matrix[i, j] = matrix[i, j] + matrix[i - 1, j];
    }
    else
    {
        matrix[i, j] = matrix[i, j] + matrix[i, j - 1];
    }
}
if (i == n1 - 1 && j == n2 - 1)
{
    max = matrix[i, j];
}
}
}
Console.WriteLine();
Console.WriteLine("Max = " + max);
Console.WriteLine("Min = " + min);
}
}
}

```

Блок-схема программы:





Вывод max, min

Конец