

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 ЛАБОРАТОРНОЙ РАБОТЕ №1
дисциплины «Алгоритмизация»

Выполнил:
Кенесбаев Хилол Куат улы
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Руководитель практики:
Воронкин Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Порядок выполнения работы:

Задание 16. Обозначим остаток от деления натурального числа a на натуральное число b как $a \bmod b$. Алгоритм вычисления значения функции $F(n)$, где n — целое неотрицательное число, задан следующими соотношениями:

$$F(0) = 0;$$

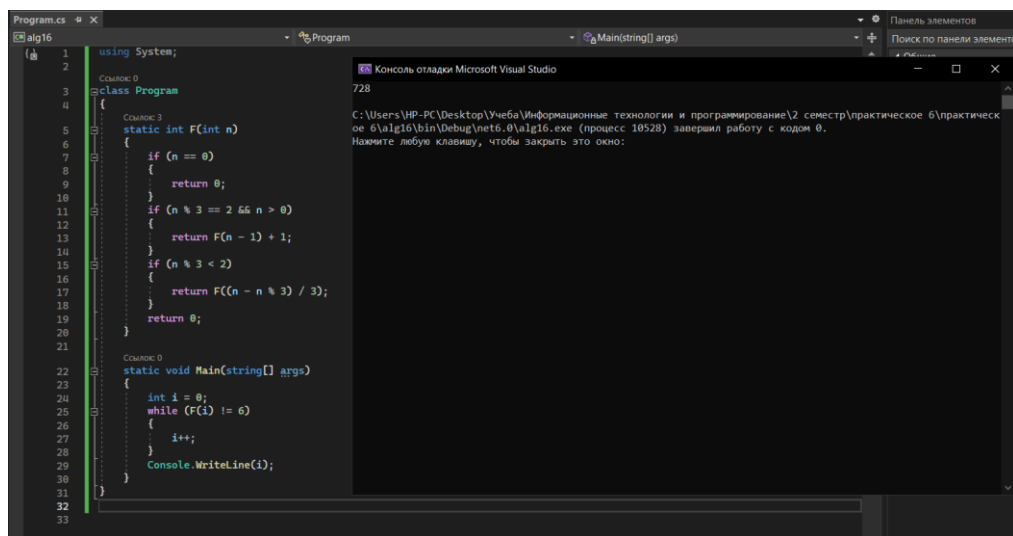
$$F(n) = F(n - 1) + 1, \text{ если } n > 0 \text{ и при этом } n \bmod 3 = 2;$$

$$F(n) = F((n - n \bmod 3) / 3), \text{ если } n > 0 \text{ и при этом } n \bmod 3 < 2.$$

Укажите наименьшее возможное n , для которого $F(n) = 6$.

№ 40732

- 1) Написал программу для вычисления значения функции $F(n)$:



```
1 using System;
2
3 class Program
4 {
5     static int F(int n)
6     {
7         if (n == 0)
8         {
9             return 0;
10        }
11        if (n % 3 == 2 && n > 0)
12        {
13            return F(n - 1) + 1;
14        }
15        if (n % 3 < 2)
16        {
17            return F((n - n % 3) / 3);
18        }
19        return 0;
20    }
21
22    static void Main(string[] args)
23    {
24        int i = 0;
25        while (F(i) != 6)
26        {
27            i++;
28        }
29        Console.WriteLine(i);
30    }
31 }
32
33
```

Консоль отладки Microsoft Visual Studio

728

C:\Users\HP-PC\Desktop\Учеба\Информационные технологии и программирование\2 семестр\практическое 6\практическое 6\algr16\bin\Debug\net6.0\algr16.exe (процесс 10528) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:

Рисунок 1. Код и результат программы algr16

- 2) Составил блок-схему программы:

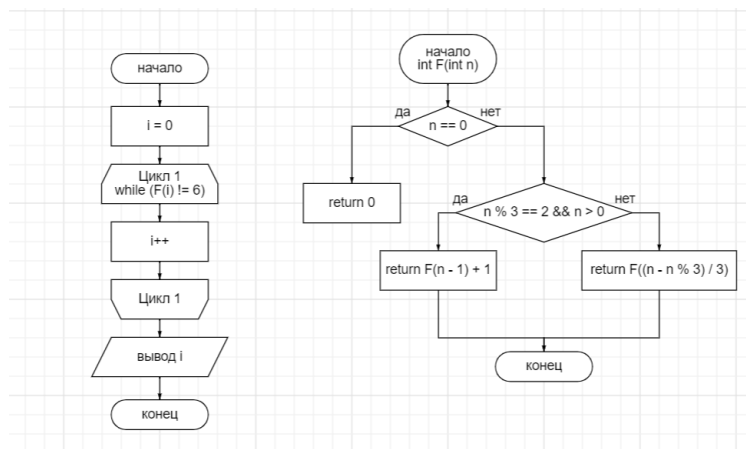
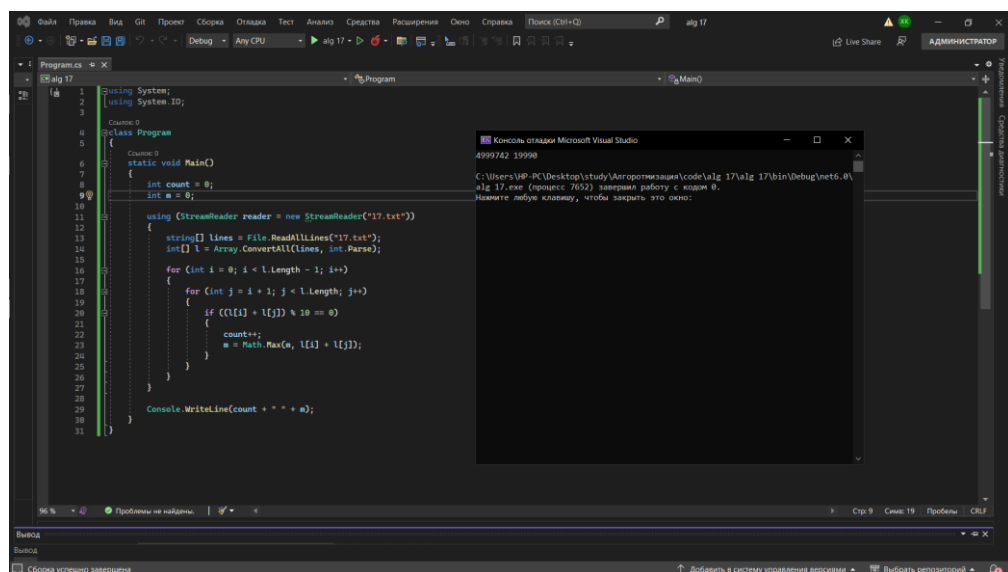


Рисунок 2. Блок-схема программы algr16

Задание 17. В файле содержится последовательность из 10 000 целых положительных чисел. Каждое число не превышает 10 000. Определите и запишите в ответе сначала количество пар элементов последовательности, у которых сумма элементов кратна 10, затем максимальную из сумм элементов таких пар. В данной задаче под парой подразумевается два различных элемента последовательности. Порядок элементов в паре не важен.

№ 37358

1) Написал программу, которая определяет количество пар элементов последовательности, у которых сумма элементов кратна 10 затем максимальную из сумм элементов таких пар:



```
1  using System;
2  using System.IO;
3
4  class Program
5  {
6      static void Main()
7      {
8          int count = 0;
9          int m = 0;
10
11          using (StreamReader reader = new StreamReader("17.txt"))
12          {
13              string[] lines = File.ReadAllLines("17.txt");
14              int[] l = Array.ConvertAll(lines, int.Parse);
15
16              for (int i = 0; i < l.Length - 1; i++)
17              {
18                  for (int j = i + 1; j < l.Length; j++)
19                  {
20                      if ((l[i] + l[j]) % 10 == 0)
21                      {
22                          count++;
23                          m = Math.Max(m, l[i] + l[j]);
24                      }
25                  }
26              }
27
28              Console.WriteLine(count + " " + m);
29          }
30      }
31  }
```

Консоль отладки Microsoft Visual Studio

4999742 19998

C:\Users\198-PC\Desktop\study\Алгоритмы\algr17\algr17\bin\Debug\net6.0\algr17.exe (процесс 7632) завершила работу с кодом 0.

Нажмите любую клавишу, чтобы закрыть это окно:

Рисунок 3. Код и результат программы algr17

2) Составил блок-схему программы:

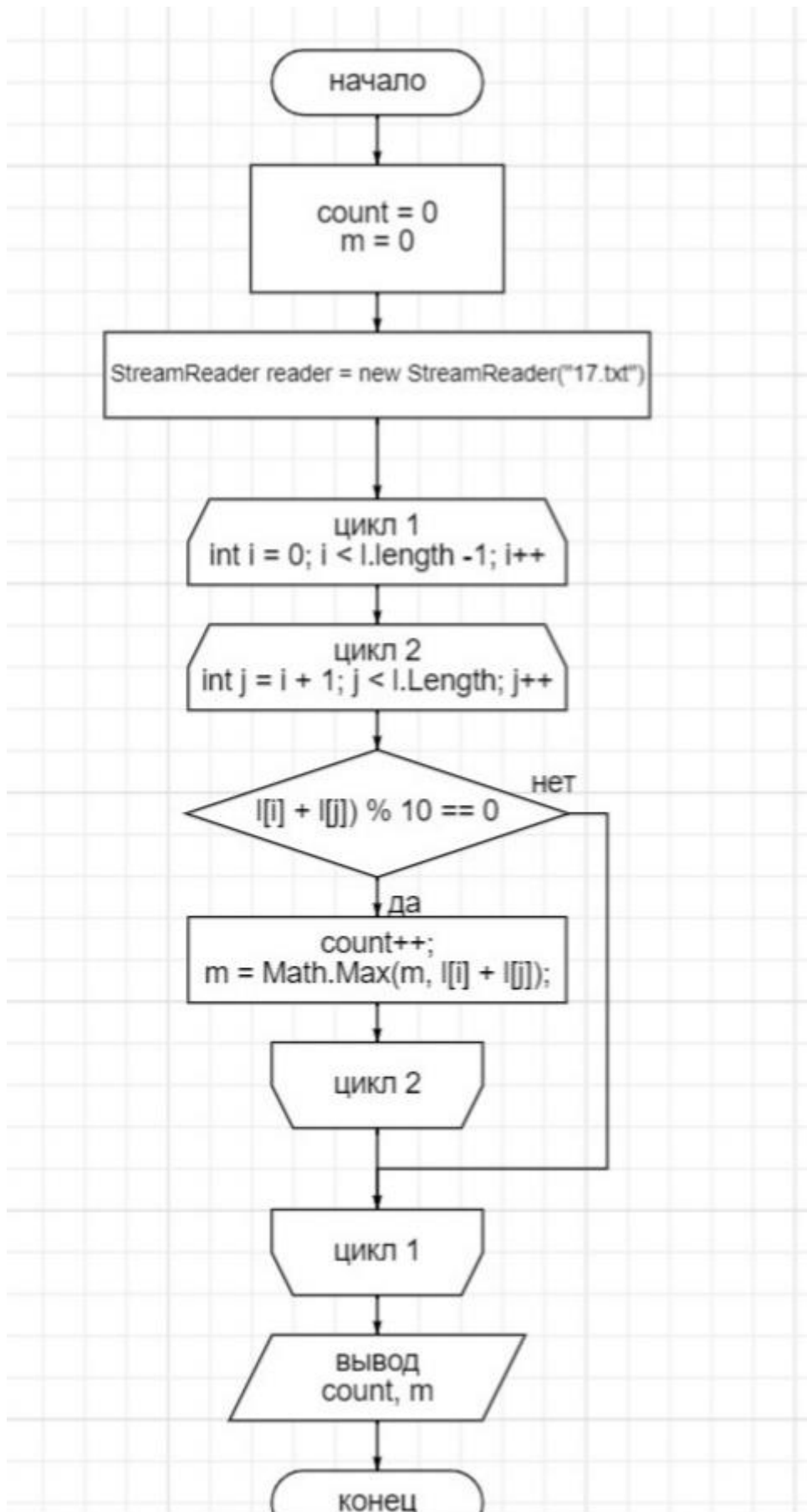


Рисунок 4. Блок-схема программы algr17

Задание 18. Квадрат разлинован 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.

Написал программу, которая определяет максимальную денежную сумму, которую может собрать Робот, пройдя из левой нижней клетки таблицы в правую верхнюю:

```

1 // alg 18
2 #include <iostream>
3 using namespace std;
4 int main() {
5     setlocale(LC_ALL, "ru");
6     int array[16][16] = {
7         {86, 40, 41, 13, 91, 96, 88, 42, 73, 51},
8         {106, 23, 6, 19, 54, 66, 63, 52, 51, 12},
9         {58, 80, 49, 30, 10, 22, 30, 62},
10        {82, 99, 97, 83, 92, 52, 64, 58, 5, 53},
11        {22, 77, 78, 43, 62, 86, 27, 3, 28, 53},
12        {21, 7, 99, 20, 6, 62, 30, 45, 18, 30},
13        {33, 80, 43, 99, 97, 71, 70, 40, 20, 89},
14        {13, 43, 83, 20, 64, 53, 11, 24, 72, 85},
15        {129, 77, 81, 91, 41, 9, 1, 57, 86, 45},
16        {17, 69, 29, 85, 33, 69, 100, 69, 19, 75}
17    };
18
19    // Вывод исходной таблицы
20    cout << "Исходная таблица:" << endl;
21    for (int i = 0; i < 16; i++)
22    {
23        for (int j = 0; j < 16; j++)
24        {
25            cout << array[i][j] << " ";
26        }
27        cout << endl;
28    }
29
30    // Сложение значений по горизонтали
31    for (int i = 1; i < 16; i++)
32    {
33        array[i][0] = array[i - 1][0];
34    }
35
36    // Сложение значений по вертикали
37    for (int j = 1; j < 16; j++)
38    {
39        array[0][j] = array[0][j - 1];
40    }
41
42    // Сложение значений по диагонали
43    for (int i = 1; i < 16; i++)
44    {
45        for (int j = 1; j < 16; j++)
46        {
47            int maxSummet = max(array[i][j] - 1, array[i - 1][j]);
48            array[i][j] = maxSummet;
49        }
50    }
51
52    // Вывод измененной таблицы
53    cout << endl << "Измененная таблица:" << endl;
54    for (int i = 0; i < 16; i++)
55    {
56        for (int j = 0; j < 16; j++)
57        {
58            cout << array[i][j] << " ";
59        }
60        cout << endl;
61    }
62 }

```

Исходная таблица:

86	40	41	13	91	96	88	42	73	51
106	23	6	19	54	66	63	52	51	12
58	80	49	30	10	22	30	62		
82	99	97	83	92	52	64	58	5	53
22	77	78	43	62	86	27	3	28	53
21	7	99	20	6	62	30	45	18	30
33	80	43	99	97	71	70	40	20	89
13	43	83	20	64	53	11	24	72	85
129	77	81	91	41	9	1	57	86	45
17	69	29	85	33	69	100	69	19	75

Измененная таблица:

86	126	167	180	271	367	455	497	570	621
182	205	211	230	329	447	518	550	601	622
280	366	455	515	551	646	681	703	733	815
362	405	562	645	737	789	853	911	916	969
384	542	632	688	829	915	942	945	973	1024
405	549	731	751	835	977	1007	1050	1068	1102
440	629	774	873	970	1048	1123	1163	1202	1291
455	672	857	893	1064	1097	1124	1187	1274	1376
494	740	938	1020	1105	1114	1135	1244	1360	1421
511	818	967	1114	1147	1216	1316	1385	1484	1496

Рисунок 5. Код и результат программы algr18

1) Составил блок-схему программы:

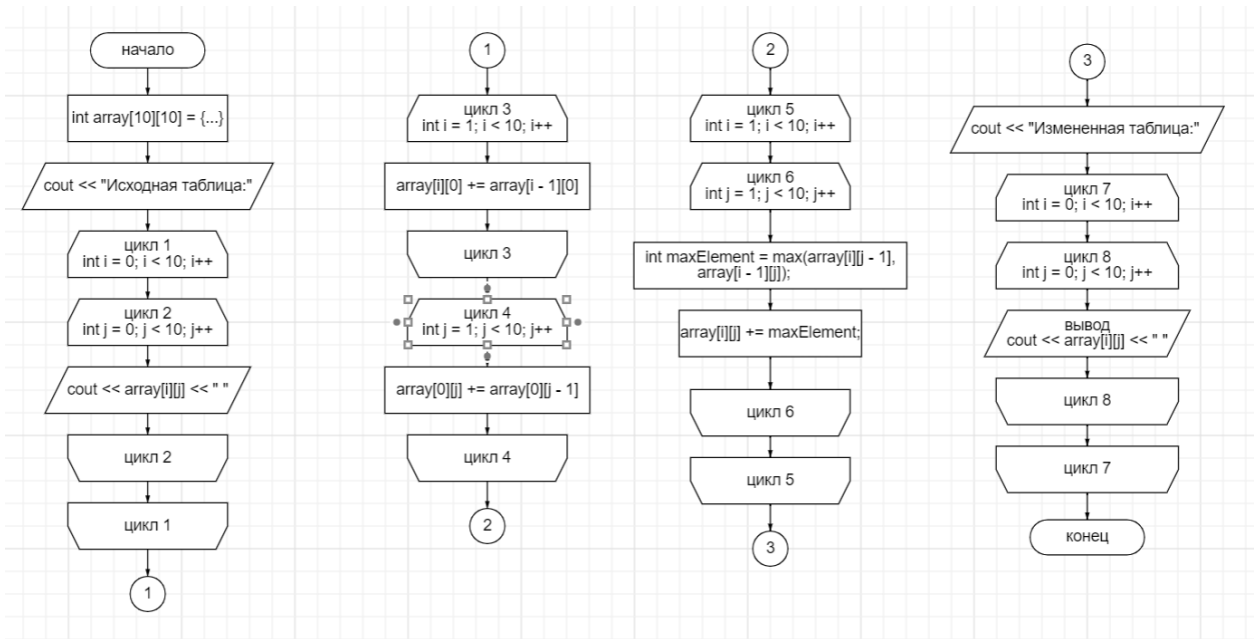


Рисунок 6. Блок-схема программы algr18

2) Далее написал программу, которая определяет уже минимальную денежную сумму, которую может собрать Робот, пройдя из левой нижней клетки таблицы в правую верхнюю:

```

// algr18.cpp
// Исходная таблица
int array[10][10] = {
    {86, 40, 41, 13, 91, 96, 88, 42, 73, 51},
    {96, 23, 6, 19, 58, 89, 43, 32, 31, 1},
    {98, 86, 89, 60, 36, 95, 35, 22, 30, 82},
    {82, 99, 97, 83, 92, 52, 64, 58, 5, 53},
    {22, 77, 70, 43, 92, 86, 27, 3, 20, 51},
    {21, 7, 99, 20, 6, 62, 30, 43, 18, 34},
    {35, 80, 43, 99, 97, 71, 75, 40, 39, 89},
    {15, 43, 83, 20, 94, 33, 11, 24, 72, 85},
    {39, 77, 81, 91, 41, 9, 1, 57, 86, 45},
    {17, 69, 29, 85, 33, 69, 100, 69, 19, 75}
};

// Вывод исходной таблицы
cout << "Исходная таблица:" << endl;
for (int i = 0; i < 10; i++)
{
    for (int j = 0; j < 10; j++)
    {
        cout << array[i][j] << " ";
    }
    cout << endl;
}

// Сложение элементов по вертикали
for (int i = 1; i < 10; i++)
{
    array[i][0] += array[i - 1][0];
}

// Сложение элементов по горизонтали
for (int j = 1; j < 10; j++)
{
    array[0][j] += array[0][j - 1];
}

// Сложение элементов по диагонали
for (int i = 1; i < 10; i++)
{
    for (int j = 1; j < 10; j++)
    {
        int maxElement = min(array[i][j] - 1, array[i - 1][j]);
        array[i][j] += maxElement;
    }
}

// Вывод измененной таблицы
cout << "Измененная таблица:" << endl;
for (int i = 0; i < 10; i++)
{
    for (int j = 0; j < 10; j++)
    {
        cout << array[i][j] << " ";
    }
    cout << endl;
}

return 0;
  
```

Консоль отладки Microsoft Visual Studio

Исходная таблица:

```

86 40 41 13 91 96 88 42 73 51
96 23 6 19 58 89 43 32 31 1
98 86 89 60 36 95 35 22 30 82
82 99 97 83 92 52 64 58 5 53
22 77 70 43 92 86 27 3 20 51
21 7 99 20 6 62 30 43 18 34
35 80 43 99 97 71 75 40 39 89
15 43 83 20 94 33 11 24 72 85
39 77 81 91 41 9 1 57 86 45
17 69 29 85 33 69 100 69 19 75
  
```

Измененная таблица:

```

86 126 167 180 271 367 455 497 570 621
182 149 155 174 232 312 375 407 438 439
280 235 244 234 268 363 398 420 450 521
362 334 341 317 360 412 462 478 455 508
384 411 411 360 452 498 480 481 483 534
405 412 510 380 386 448 478 521 501 535
440 492 535 479 483 519 553 561 540 624
455 498 581 499 577 552 563 585 612 697
494 571 652 590 610 561 562 619 698 742
511 580 689 675 651 630 662 688 707 782
  
```

C:\Users\HP-PC\Desktop\study\Алгоритмизация\code\alg 18\alg 18\Debug\alg 18.exe (процесс 18668) завершила работу с кодом 0.

Нажмите любую клавишу, чтобы закрыть это окно:

Рисунок 7. Код и результат программы algr18

3) Составил блок-схему программы

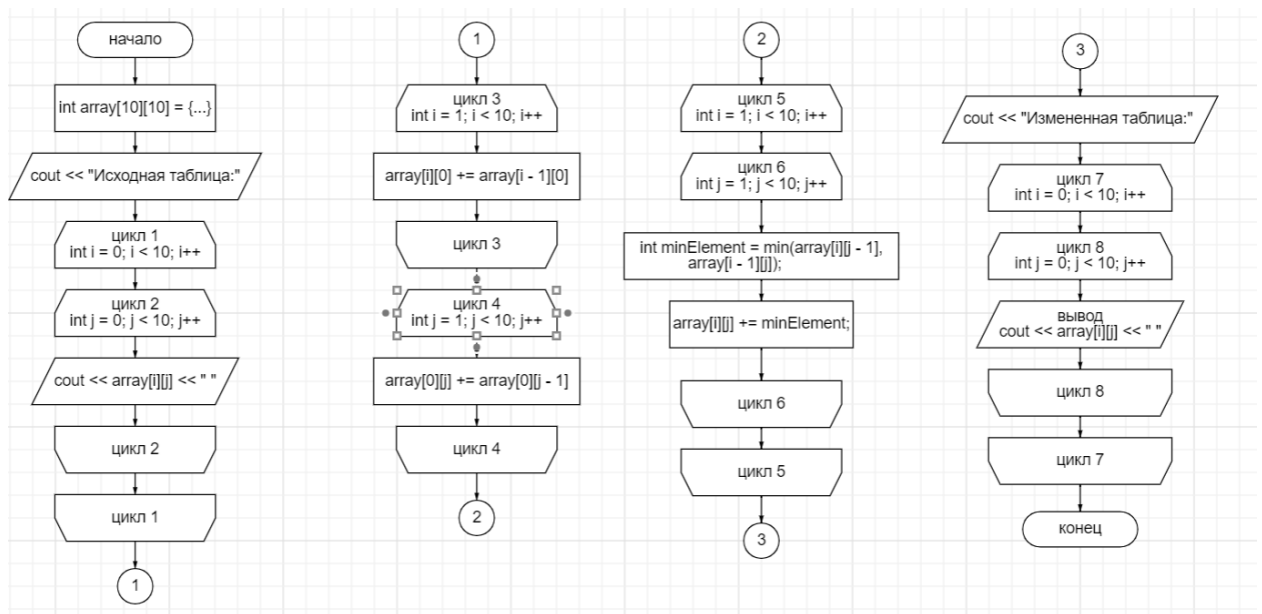


Рисунок 8. Блок-схема программы algr18