

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:
Зармухамбетов Булат
Эльдарович
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)

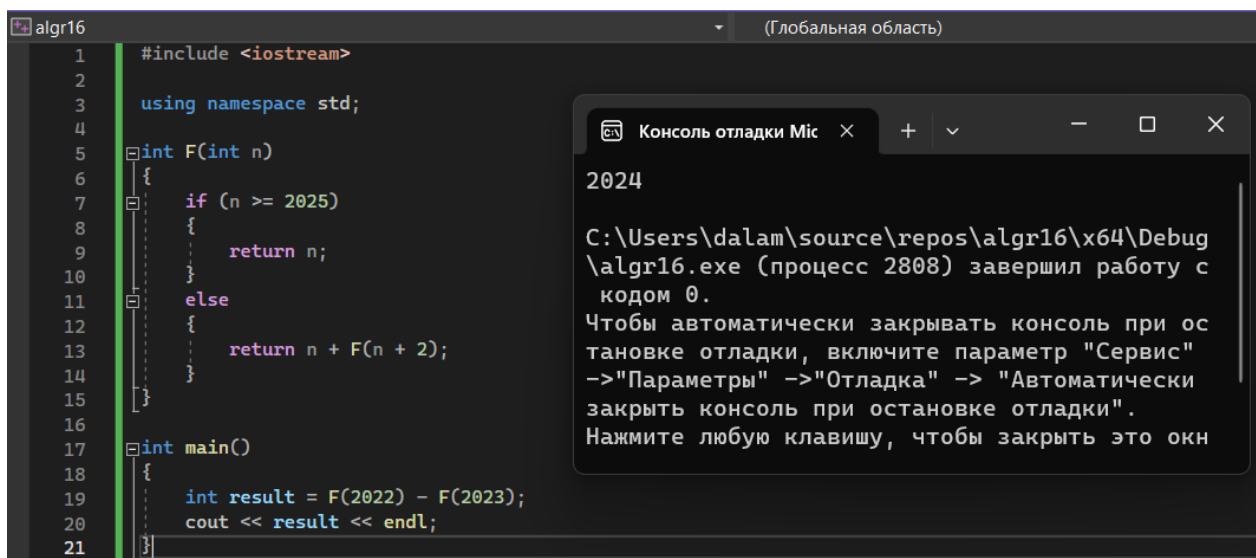
Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Порядок выполнения работы:

Задание 16. Алгоритм вычисления значения функции $F(n)$, где n – натуральное число, задан следующими соотношениями: $F(n) = n$, если $n \geq 2025$, $F(n) = n + F(n + 2)$, если $n < 2025$. Чему равно значение выражения $F(2022) - F(2023)$? Номер задания 57423.

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



```
1 #include <iostream>
2
3 using namespace std;
4
5 int F(int n)
6 {
7     if (n >= 2025)
8     {
9         return n;
10    }
11    else
12    {
13        return n + F(n + 2);
14    }
15 }
16
17 int main()
18 {
19     int result = F(2022) - F(2023);
20     cout << result << endl;
21 }
```

Консоль отладки Mic x + - □ x

2024

C:\Users\dalam\source\repos\algr16\x64\Debug\algr16.exe (процесс 2808) завершил работу с кодом 0.

Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно

Рисунок 1. Код и результат программы algr16

- 2) Составил блок-схему программы:

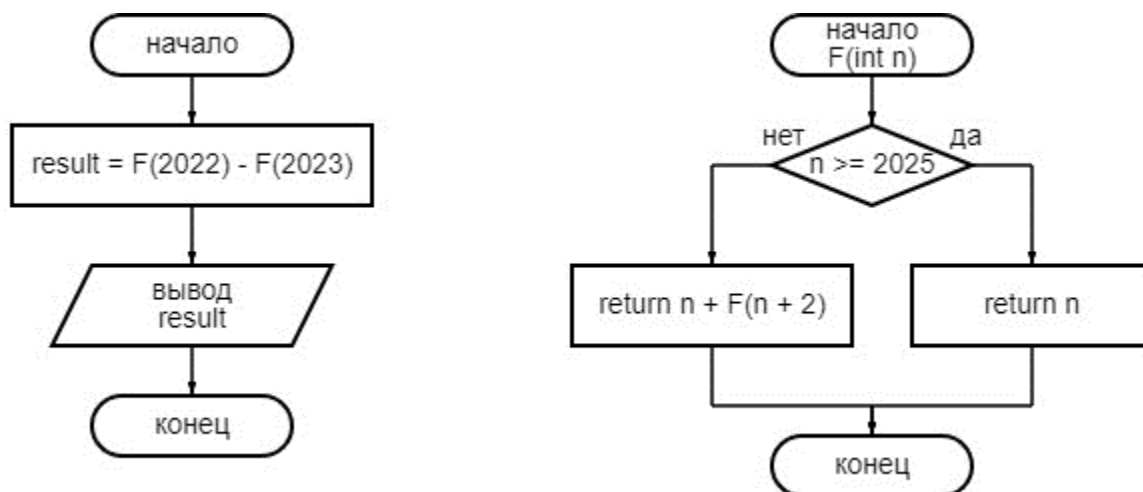


Рисунок 2. Блок-схема программы algr16

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, для

которых произведение элементов делится без остатка на 10, затем максимальную из сумм элементов таких пар. Номер задания 37344.

1) Написал программу, которая определяет количество пар элементов, которые делятся без остатка на 10, и определяет максимальную сумму таких пар:

The screenshot shows a C++ program in a code editor and its execution in a console window. The program reads numbers from a file '17.txt' and finds pairs of numbers whose product is divisible by 10, tracking the count of such pairs and the maximum sum of such pairs.

```

1  #include <iostream>
2  #include <fstream>
3
4  using namespace std;
5
6  int main()
7  {
8      ifstream f("17.txt");
9      int count = 0;
10     int maxsum = 0;
11     int arr[10000];
12
13     for (int i = 0; i < 10000; i++)
14     {
15         f >> arr[i];
16     }
17
18     for (int i = 0; i < 9999; i++)
19     {
20         for (int j = i + 1; j < 10000; j++)
21         {
22             if ((arr[i] * arr[j]) % 10 == 0)
23             {
24                 count++;
25             }
26             if (arr[i] + arr[j] > maxsum)
27             {
28                 maxsum = arr[i] + arr[j];
29             }
30         }
31     }
32     cout << count << " " << maxsum << endl;
33     return 0;
34 }

```

The console output shows the result: 13510315 19999. Below the output, there is a message from the Microsoft Visual Studio debugger stating that the program has finished with code 0 and provides instructions on how to automatically close the console window.

Рисунок 3. Код и результат программы algr17

2) Составил блок-схему программы:

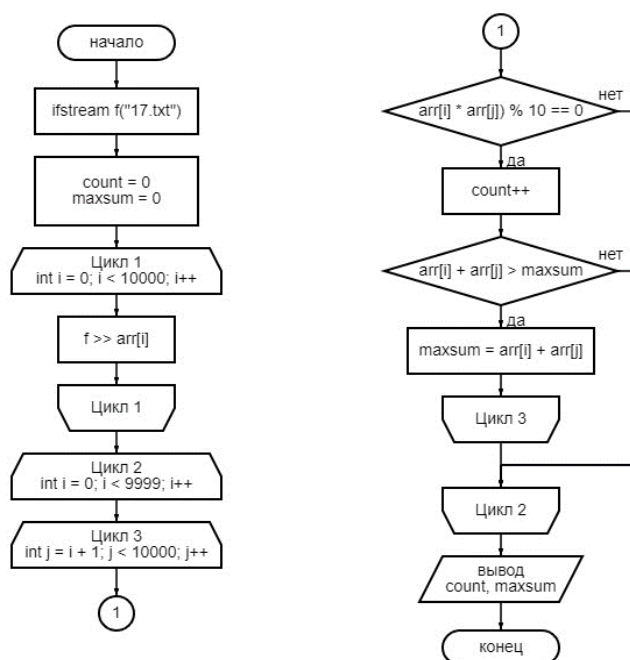


Рисунок 4. Блок-схема программы algr17

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 вверх. По команде вправо Робот перемещается в соседнюю правую клетку, по команде вверх – в соседнюю верхнюю. 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. Определите максимальную и минимальную денежную сумму, которую может собрать Робот, пройдя из левой нижней клетки в правую верхнюю. Номер задания 27668.

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

```

1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     setlocale(LC_ALL, "ru");
6     int array[10][10] = {
7         {23, 96, 3, 62, 59, 10, 73, 30, 10, 81},
8         {44, 3, 99, 88, 53, 90, 42, 39, 36, 98},
9         {39, 32, 5, 8, 34, 9, 39, 95, 36, 41},
10        {33, 79, 22, 99, 29, 61, 65, 48, 70, 77},
11        {73, 49, 40, 39, 16, 49, 68, 37, 60, 10},
12        {98, 65, 67, 11, 14, 39, 74, 29, 70, 81},
13        {29, 90, 63, 44, 5, 4, 5, 28, 22, 94},
14        {88, 100, 15, 75, 42, 68, 50, 8, 98, 55},
15        {38, 21, 77, 25, 91, 92, 64, 72, 16, 67},
16        {65, 93, 45, 52, 51, 19, 96, 45, 7, 39}
17    };
18    // Вывод матрицы
19    cout << "Исходная таблица:" << endl;
20    for (int i = 0; i < 10; i++)
21    {
22        for (int j = 0; j < 10; j++)
23        {
24            cout << array[i][j] << " ";
25        }
26        cout << endl;
27    }
28    // Сложение элементов по вертикали
29    for (int i = 8; i >= 0; i--)
30    {
31        array[i][0] += array[i + 1][0];
32    }
33    // Сложение элементов по горизонтали
34    for (int j = 1; j < 10; j++)
35    {
36        array[9][j] += array[9][j - 1];
37    }
38    // Сложение элементов по диагонали
39    for (int i = 8; i >= 0; i--)
40    {
41        for (int j = 1; j < 10; j++)
42        {
43            int maxElement = max(array[i][j-1], array[i+1][j]);
44            array[i][j] += maxElement;
45        }
46    }
47    // Вывод измененной матрицы
48    cout << endl << "Измененная таблица:" << endl;
49    for (int i = 0; i < 10; i++)
50    {
51        for (int j = 0; j < 10; j++)
52        {
53            cout << array[i][j] << " ";
54        }
55        cout << endl;
56    }
57 }

```

Консоль отладки Micro:

Исходная таблица:

```

23 96 3 62 59 10 73 30 10 81
44 3 99 88 53 90 42 39 36 98
39 32 5 8 34 9 39 95 36 41
33 79 22 99 29 61 65 48 70 77
73 49 40 39 16 49 68 37 60 10
98 65 67 11 14 39 74 29 70 81
29 90 63 44 5 4 5 28 22 94
88 100 15 75 42 68 50 8 98 55
38 21 77 25 91 92 64 72 16 67
65 93 45 52 51 19 96 45 7 39

```

Измененная таблица:

```

530 705 713 860 919 951 1056 1086 1096 1276
507 609 710 798 851 941 983 1032 1068 1195
463 606 611 703 758 794 889 993 1029 1097
424 574 596 695 724 785 850 898 968 1056
391 495 553 592 608 657 754 791 891 979
318 446 513 524 538 600 686 715 831 969
220 381 444 488 493 561 612 661 761 888
191 291 306 381 439 557 607 633 739 794
103 179 280 305 397 489 553 625 641 708
65 158 203 255 306 325 421 466 473 512

```

C:\Users\dalam\source\repos\algr18\x64\Debug\algr18.exe (процесс 11396) завершил работу с кодом 0.

Чтобы автоматически закрывать консоль при отладке, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически зак

Рисунок 5. Код и результат программы algr18

2) Составил блок-схему программы:

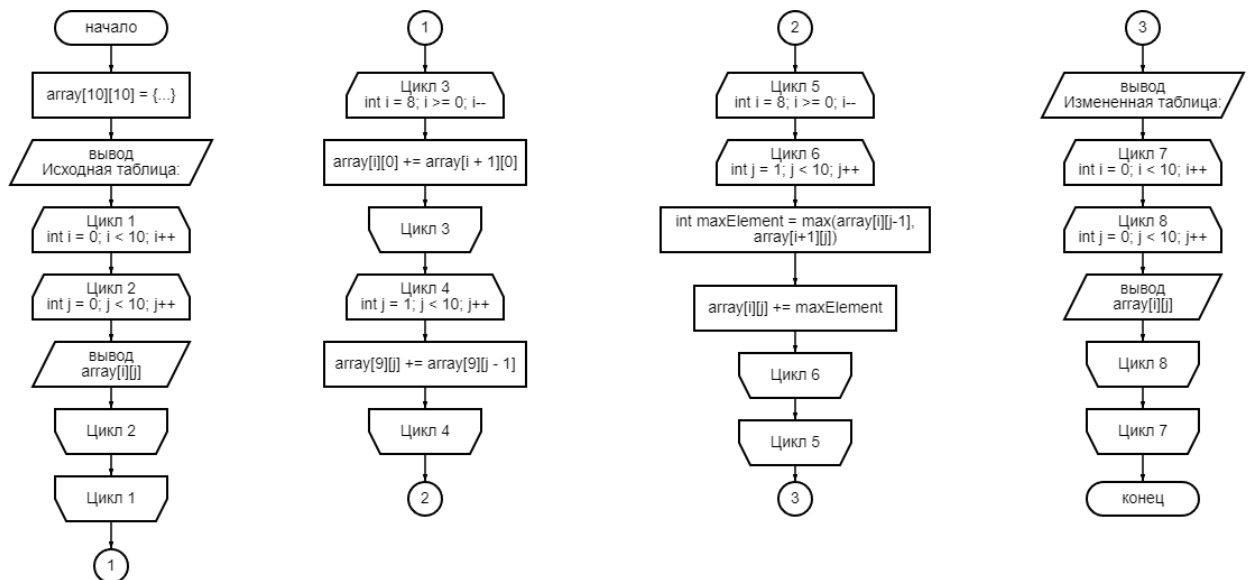


Рисунок 6. Блок-схема программы algr18

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

```

algr18
#include <iostream>
using namespace std;

int main() {
    setlocale(LC_ALL, "ru");
    int array[10][10] = {
        {23, 96, 3, 62, 59, 10, 73, 30, 10, 81},
        {44, 3, 99, 88, 53, 90, 42, 39, 36, 98},
        {39, 32, 5, 8, 34, 9, 39, 95, 36, 41},
        {33, 79, 22, 99, 29, 61, 65, 48, 70, 77},
        {73, 49, 40, 39, 16, 49, 68, 37, 60, 10},
        {98, 65, 67, 11, 14, 39, 74, 29, 70, 81},
        {29, 90, 63, 44, 5, 4, 5, 28, 22, 94},
        {80, 100, 15, 75, 42, 68, 50, 8, 90, 55},
        {38, 21, 77, 25, 91, 92, 64, 72, 16, 67},
        {65, 93, 45, 52, 51, 19, 96, 45, 7, 39}
    };

    // Вывод матрицы
    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 = 8; i >= 0; i--)
    {
        array[i][0] += array[i + 1][0];
    }

    // Сложение элементов по горизонтали
    for (int j = 1; j < 10; j++)
    {
        array[9][j] += array[9][j - 1];
    }

    // Сложение элементов по диагонали
    for (int i = 8; i >= 0; i--)
    {
        for (int j = 1; j < 10; j++)
        {
            int minElement = min(array[i][j-1], array[i+1][j]);
            array[i][j] += minElement;
        }
    }

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

Исходная таблица:

```

23 96 3 62 59 10 73 30 10 81
44 3 99 88 53 90 42 39 36 98
39 32 5 8 34 9 39 95 36 41
33 79 22 99 29 61 65 48 70 77
73 49 40 39 16 49 68 37 60 10
98 65 67 11 14 39 74 29 70 81
29 90 63 44 5 4 5 28 22 94
80 100 15 75 42 68 50 8 90 55
38 21 77 25 91 92 64 72 16 67
65 93 45 52 51 19 96 45 7 39
  
```

Измененная таблица:

```

530 594 515 571 533 530 584 580 590 671
507 498 512 509 474 520 511 550 586 684
463 495 413 421 421 430 469 564 585 619
424 503 408 472 387 448 513 479 549 578
391 424 386 373 358 407 475 431 491 501
318 375 346 334 342 371 411 394 457 538
220 310 279 323 328 332 337 365 387 481
191 224 216 291 333 401 451 459 557 611
103 124 201 226 317 409 473 538 489 556
65 158 203 255 306 325 421 466 473 512
  
```

C:\Users\dalam\source\repos\algr18\x64\Debug\algr18.exe (процесс 16192) завершил работу с кодом 0. Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".

Рисунок 7. Код и результат программы algr18

4) Составил блок-схему программы

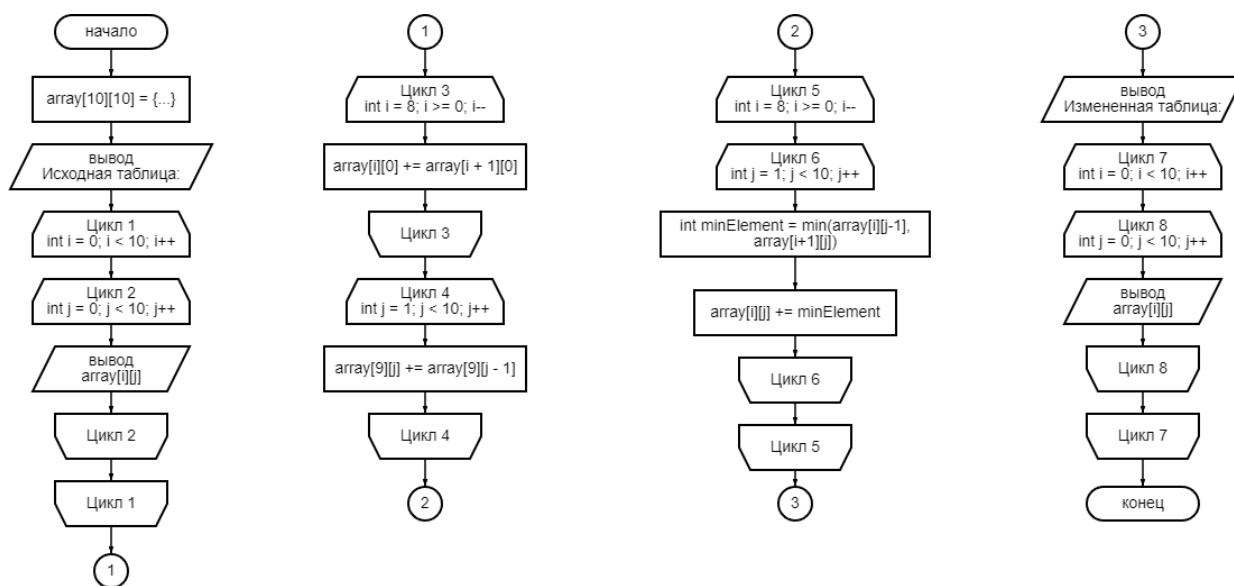


Рисунок 8. Блок-схема программы `algr18`