

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:
Епифанов Алексей Александрович
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:

1. Выполнил 16 задание под номером 59721

11

Тип 16 № 59721

Алгоритм вычисления значения функции $F(n)$, где n — натуральное число, задан следующими соотношениями:
 $F(n) = n$, если $n = 1$;
 $F(n) = n - 1 + F(n - 1)$, если $n > 1$.

Чему равно значение выражения $F(2024) - F(2022)$?

Источник: ЕГЭ по информатике 19.06.2023. Основная волна

Решение · Подготовка к ЕГЭ-2024 · Помощь

Рисунок 1. Задание

```
program > G+ alg.cpp > ...
1  #include <iostream>
2  #include <math.h>
3  using namespace std;
4  int func(int n)
5  {
6      if (n == 1)
7          return n;
8      else
9          return n - 1 + func(n - 1);
10 }
11 int main()
12 {
13     int a = func(2024) - func(2022);
14     cout << a << endl;
15 }
16
```

output — zsh — 80x
Last login: Sun Sep 10 17:32:59 on ttys027
./"alg"
aleksejepifanov@MacBook-Pro output % ./"alg"
4045
aleksejepifanov@MacBook-Pro output %

Рисунок 2. Код и работа программы alg

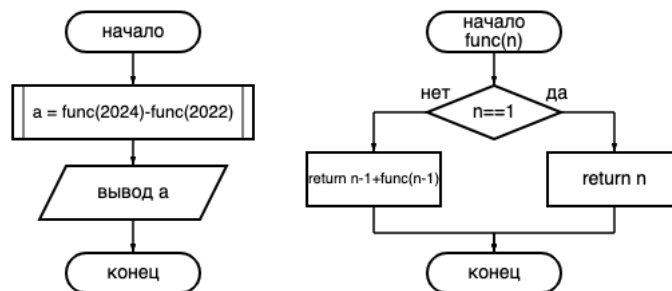


Рисунок 3. Блок-схема программы alg

2. Выполнил 17 задание под номером 59784

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.

Ответ:

Источник: ЕГЭ по информатике 19.06.2023. Основная волна. Разные города

Решение · [Подготовка к ЕГЭ-2024](#) · [Помощь](#)

Рисунок 4. Задание

```
program > alg2.cpp > main()
1  #include <iostream>
2  #include <fstream>
3  #include <vector>
4  #include <math.h>
5  using namespace std;
6  int main()
7  {
8      std::ifstream inputFile("/Users/aleksejepifanov/Desktop/нары/нары_3_сем/pytgit/algorithm/program/17.txt");
9      std::vector<int> numbers;
10     int number;
11     int count = 0;
12
13     while (inputFile >> number)
14     {
15         numbers.push_back(number);
16     }
17     inputFile.close();
18     int max = 0;
19
20     for (int i = 0; i < (int)numbers.size(); i++)
21     {
22         if (numbers[i] % 100 == 15 && numbers[i] > max)
23             max = numbers[i];
24     }
25     int min_sum = max;
26
27     for (int i = 0; i <= (int)numbers.size() - 2; i++)
28     {
29         int c = 0, sum = 0;
30         int d[] = {numbers[i], numbers[i + 1], numbers[i + 2]};
31
32         for (int j = 0; j < 3; j++)
33         {
34             sum += d[j];
35             if (abs(d[j]) > 999 && abs(d[j]) < 10000)
36                 c++;
37         }
38         if (c == 1 && sum < max)
39         {
40             count++;
41             if (sum < min_sum)
42                 min_sum = sum;
43         }
44     }
45     cout << count << "    " << min_sum << endl;
46 }
```

output — zsh — 80x24

```
Last login: Sun Sep 10 18:02:26 on ttys029
aleksejepifanov@MacBook-Pro output % ./"alg2"
2453    -176846
aleksejepifanov@MacBook-Pro output %
```

Рисунок 5. Код и работа программы alg2

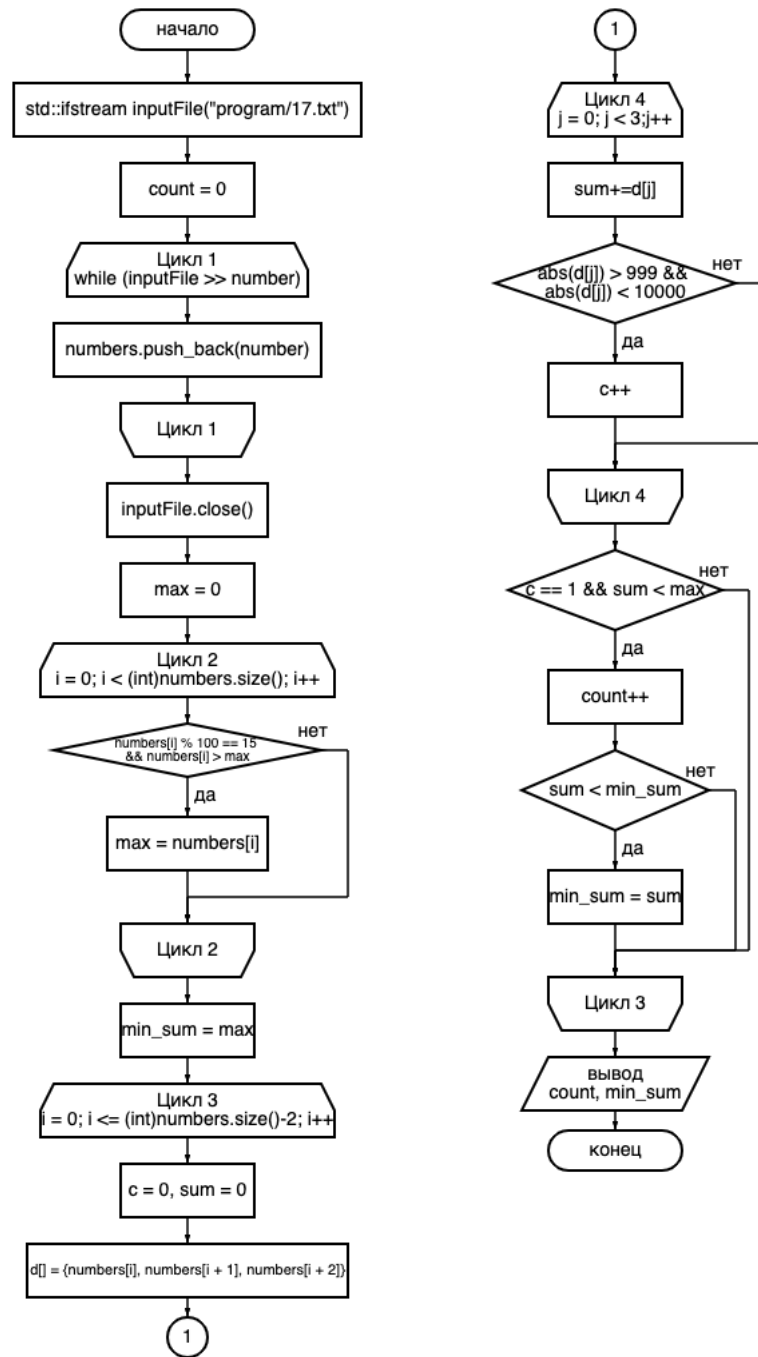


Рисунок 6. Блок-схема программы `alg2`

3. Выполнил 18 задание под номером 36873

Тип 18 № 36873

Дан квадрат 15×15 клеток, в каждой клетке которого записано целое число. В левом нижнем углу квадрата стоит робот. За один ход робот может переместиться на одну клетку вправо или на одну клетку вверх. Выходить за пределы квадрата робот не может. При этом ведётся подсчёт суммы по следующим правилам: число в очередной клетке, через которую проходит робот, включается в сумму, если оно больше числа в предыдущей клетке на пути робота. Если число в очередной клетке не больше числа в предыдущей, сумма не изменяется. Число в начальной клетке всегда включается в сумму. Необходимо переместить робота в правый верхний угол так, чтобы полученная сумма была максимальной. В ответе запишите максимально возможную сумму.

Исходные данные записаны в электронной таблице.

Задание 18

Пример входных данных (для таблицы размером 4×4):

44	42	89	37
18	35	50	20
6	41	26	64
7	9	70	85

Для указанных входных данных оптимальным маршрутом будет путь по клеткам 7, 9, 70, 26, 50, 89, 37. Итоговая сумма равна $7 + 9 + 70 + 50 + 89 = 225$. Числа 26 и 37 не включаются в сумму, так как $26 < 70$ и $37 < 89$.

[Показать решение](#)

Рисунок 7. Задание

```
program > C:\alg3.cpp > main()
8 {
9     for (int i = 0; i < 15; i++)
10     {
11         for (int j = 0; j < 15; j++)
12         {
13             std::cout << std::setw(4) << matrix[i][j] << " ";
14         }
15         std::cout << std::endl;
16     }
17 }
18 int main()
19 {
20     int matrix[15][15] = {
21         {12, 88, 57, 63, 77, 70, 6, 98, 65, 58, 19, 73, 13, 86, 1},
22         {12, 66, 3, 14, 36, 65, 51, 4, 45, 78, 12, 31, 91, 12, 86},
23         {61, 41, 46, 70, 39, 7, 76, 23, 3, 4, 62, 75, 63, 65, 54},
24         {86, 73, 65, 44, 42, 14, 50, 92, 94, 37, 6, 78, 46, 97, 17},
25         {95, 40, 51, 62, 12, 14, 58, 49, 83, 97, 70, 14, 6, 47, 55},
26         {75, 43, 22, 77, 71, 53, 19, 58, 37, 69, 48, 98, 98, 89, 35},
27         {22, 84, 34, 52, 37, 39, 72, 84, 19, 49, 10, 78, 75, 7, 42},
28         {59, 93, 9, 90, 13, 31, 93, 54, 73, 52, 4, 94, 37, 81, 22},
29         {68, 76, 84, 83, 87, 36, 26, 19, 19, 73, 24, 47, 42, 52, 5},
30         {79, 12, 76, 34, 77, 2, 70, 20, 78, 49, 7, 74, 69, 90, 45},
31         {90, 76, 32, 31, 46, 59, 35, 9, 10, 32, 93, 6, 19, 40, 60},
32         {67, 21, 74, 59, 98, 15, 49, 70, 31, 48, 63, 52, 70, 33, 69},
33         {46, 39, 51, 77, 78, 36, 46, 50, 12, 56, 75, 77, 36, 5, 11},
34         {26, 62, 85, 7, 15, 74, 39, 94, 53, 89, 53, 55, 3, 49, 4},
35         {21, 94, 40, 22, 35, 50, 85, 61, 29, 89, 18, 62, 52, 94, 16}};
36     int newmatrix[15][15];
37     newmatrix[14][0] = 21;
38     for (int i = 13; i >= 0; i--)
39     {
40         newmatrix[i][0] = newmatrix[i + 1][0] + ((matrix[i][0] > matrix[i + 1][0]) ? matrix[i][0] : 0);
41     }
42     for (int j = 1; j <= 14; j++)
43     {
44         newmatrix[14][j] = newmatrix[14][j - 1] + ((matrix[14][j] > matrix[14][j - 1]) ? matrix[14][j] : 0);
45     }
46     for (int i = 13; i >= 0; i--)
47     {
48         for (int j = 1; j <= 14; j++)
49         {
50             int a = newmatrix[i + 1][j] + ((matrix[i][j] > matrix[i + 1][j]) ? matrix[i][j] : 0);
51             int b = newmatrix[i][j - 1] + ((matrix[i][j] > matrix[i][j - 1]) ? matrix[i][j] : 0);
52             newmatrix[i][j] = max(a, b);
53         }
54     }
55     printMatrix(newmatrix);
56 }
```

```
output — -zsh — 80x24
Last login: Sun Sep 10 18:10:19 on ttys032
aleksejepifanov@MacBook-Pro output % ./alg3
420 647 647 734 844 902 902 1039 1173 1186 1205 1290 1317 1403 1403
420 559 584 671 767 832 899 941 1108 1186 1186 1217 1317 1317 1403
420 493 584 671 767 767 899 941 1063 1067 1129 1284 1226 1291 1315
420 493 584 691 767 767 823 941 1063 1066 1066 1152 1163 1261 1261
420 420 519 691 725 746 823 849 969 1066 1066 1074 1117 1164 1219
325 419 468 601 725 746 765 849 886 967 967 1074 1117 1124 1124
250 419 468 524 654 693 765 849 849 898 898 976 1019 1035 1077
250 419 434 524 617 648 741 741 814 814 814 944 944 1035 1035
250 326 434 434 617 617 617 617 678 751 800 850 864 954 954
250 250 350 350 530 530 600 600 678 678 776 850 864 954 954
250 250 274 277 453 512 512 572 582 620 776 776 795 835 895
160 160 274 277 453 453 502 572 572 620 683 683 753 753 822
93 115 200 277 355 355 401 451 451 507 582 659 659 659 670
47 115 200 200 215 289 289 383 383 472 472 527 527 576 576
21 115 115 115 150 200 285 285 285 374 374 436 436 530 530
aleksejepifanov@MacBook-Pro output %
```

Рисунок 8. Код и работа программы alg3

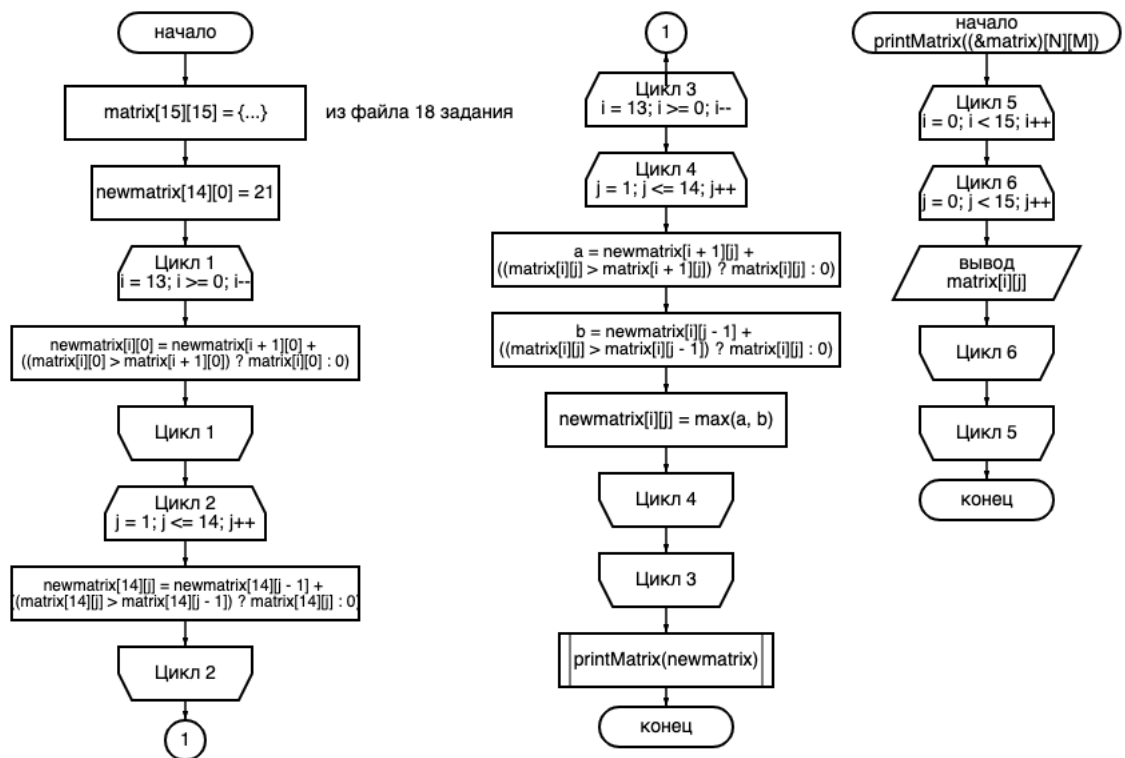


Рисунок 9. Блок-схема программы alg3