

1

{

File: right\_d.dpr

Compiler : Borland Delphi 7.0

Задача. Только вправо или вниз

-----

(Время: 1 сек. Память: 16 Мб Сложность: 32%)

Игровое поле N x M заполняется целыми числами, одно неотрицательное целое число в каждой клетке. Цель игры состоит в том, чтобы пройти по любому разрешенному пути от верхнего левого угла до правого нижнего. Целое число в каждой клетке указывает, какой длины шаг должен быть из текущей клетки. Все шаги могут быть или направо или вниз. Если в результате какого-либо шага игрок покидает пределы поля, такой шаг запрещается.

На рис. 1 приведен пример игрового поля 3 x 4, где сплошная окружность показывает положение начала (см. условие задачи в сборнике), а пунктирная окружность - цель. Рис. 2 показывает три возможных пути от начала до цели для рассматриваемого примера игрового поля, с удаленными промежуточными числами.

2-	1	1	2	2-	_	_	_	2-	_	1=	_	2-	_	1=	2=
3	2	1	44	_	_	_	_	_	_	1=	_	_	_	_	_
3	1	1	0=	3=	_	_	0=	_	_	1=	0=	_	_	_	0=

Рис. 1

Рис. 2

[

Обозначения в листинге.

Если после числа стоит символ (-), то это означает, что число обведено сплошной окружностью,

а если символ (=) - то пунктирной окружностью.

Символ (\_) - это пустая клетка

]

Требуется написать программу, которая определит число различных вариантов путей от верхнего левого угла до правого нижнего.

Входные данные

Входной файл input.txt содержит в первой строке размеры поля N (1 <= N <= 70) и M (1 <= M <= 70). В последующих N строках входного файла, каждая из которых описывает отдельную строку игрового поля, записаны через пробел по

M целых чисел – длины шагов из клеток данной строки.

Выходные данные

Выходной файл output.txt должен содержать одно число – число различных вариантов путей от верхнего левого угла до правого нижнего. Для каждого поля будет менее чем  $2^{31}$  различных путей.

Пример

N	input.txt	output.txt
1	1 4	3
	2 1 1 2	
	3 2 1 44	
	3 1 1 0	

Решение. Калмыков Вадим (ProCrypt),  
г. Сургут, ЦНИТ "Северная Звезда",  
02:47, 14.06.2008

Source : <http://acm.dvpion.ru>

Editor. MSP, 17.12.09, 11:51; LIST 6.0  
MSP, 27.09.14, 09:46

В этот листинг внесен мною текст условия задачи.

}

```
62  {$R+}  
63  {$E+}  
64  {$Q+}
```

```
65  Program RightOrDown;
```

```
66  Const  
67      Size = 70;
```

```
68  Var  
69      // числа, записанные в клетках  
70      A : array [1..Size, 1..Size] of LongInt;  
71
```

```
72      // D [i] [j] содержит число способов, которыми можно попасть из  
73      // клетки с координатами (1;1) в клетку (i;j)  
74      D : Array [1..Size, 1..Size] of LongInt;
```

```
75      i, j,          // счетчики циклов  
76      N, M : Word; // размеры поля
```

```

77 // процедура вычисления, сколькими способами можно попасть в клетку
78 // из клетки (i;j)
79 Procedure PathToCage ();

80 begin
81 // i, j - координаты клетки, из которой делаем ход

82 // если не выходим за границу строки
83 if (j + A [i] [j] <= M)
84 then
85 Inc (D [i] [j + A [i] [j]], D [i] [j]);

86 // если не выходим за границу столбца
87 if (i + A [i] [j] <= N)
88 then
89 Inc (D [i + A [i] [j]] [j], D [i] [j]);
90 end;

91 Begin
92 Assign (Input, 'input.txt');
93 Assign (Output, 'output.txt');

94 // инициализируем массив
95 for i := 1 to Size do
96 for j := 1 to Size do
97 D [i] [j] := 0;

98 Reset (Input);
99 ReadLn (N, M);

100 // считываем данные
101 for i := 1 to N do
102 for j := 1 to M do
103 Read (A [i] [j]);

104 Close (Input);
105 Rewrite (Output);

106 D [1] [1] := 1;
107 // для каждого элемента матрицы вызываем процедуру PathToCage
108 for i := 1 to N do
109 for j := 1 to M do
110 // если в данную клетку можно попасть, то необходимо искать
111 // далее
112 if (D [i] [j] <> 0)
113 and
114 (
115 (i <> N)
116 or

```

```
117         (j <> M)
118     )
119     then
120         PathToCage ();

121     Write (D [N] [M]);

122     Close (Output);
123 End.
```

Listing данной задачи опубликован в сети Internet по адресу  
<http://www.Best-Listing.ru/color-10-task-698.html>

Sergey Mitrofanov, 27.09.14, 13:03

E-mail: [infostar@mail.ru](mailto:infostar@mail.ru)

© <http://www.Best-Listing.ru/>, 2006–2014