Java May'19 DSA Recursion 2 - 1 day 21:40:06

Scrooge McDuck

Scrooge McDuck likes his treasure very much. That is why he likes to play a funny game.

He builds a labyrinth of coins and tries to escape from it. You can think of the labyrinth as a rectangular field. Each cell of the field contains 0 or more coins.

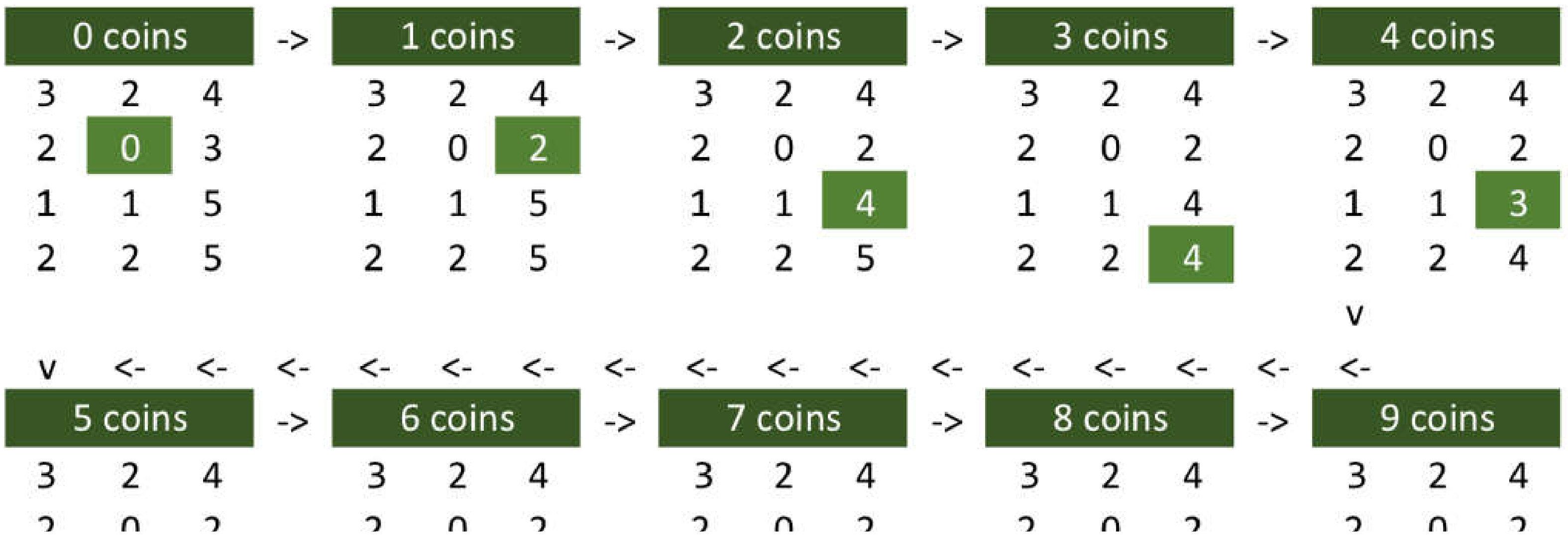
When Scrooge McDuck steps on a cell, he can take only a single coin from this cell, and only if there are any coins. Scrooge McDuck can escape the field, only if he is surrounded by empty cells.

Scrooge McDuck always wants to go to the neighbouring cell with most coins. BUT if there are more than one cells with the same amount of coins (the largest), he chooses a cell (always the largest) from the order **left, right, up, down**

If Scrooge McDuck cannot go in any direction, he is out of the labyrinth

Examples

Sample test



```
0 coins  ->  1 coins  ->  2 coins  ->  3 coins  ->  4 coins
3  2  4      3  2  4      3  2  4      3  2  4      3  2  4
2  0  3      2  0  2      2  0  2      2  0  2      2  0  2
1  1  5      1  1  5      1  1  4      1  1  4      1  1  3
2  2  5      2  2  5      2  2  5      2  2  4      2  2  4
                                                    v
v  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-
5 coins  ->  6 coins  ->  7 coins  ->  8 coins  ->  9 coins
3  2  4      3  2  4      3  2  4      3  2  4      3  2  4
2  0  2      2  0  2      2  0  2      2  0  2      2  0  2
```

Submit solution

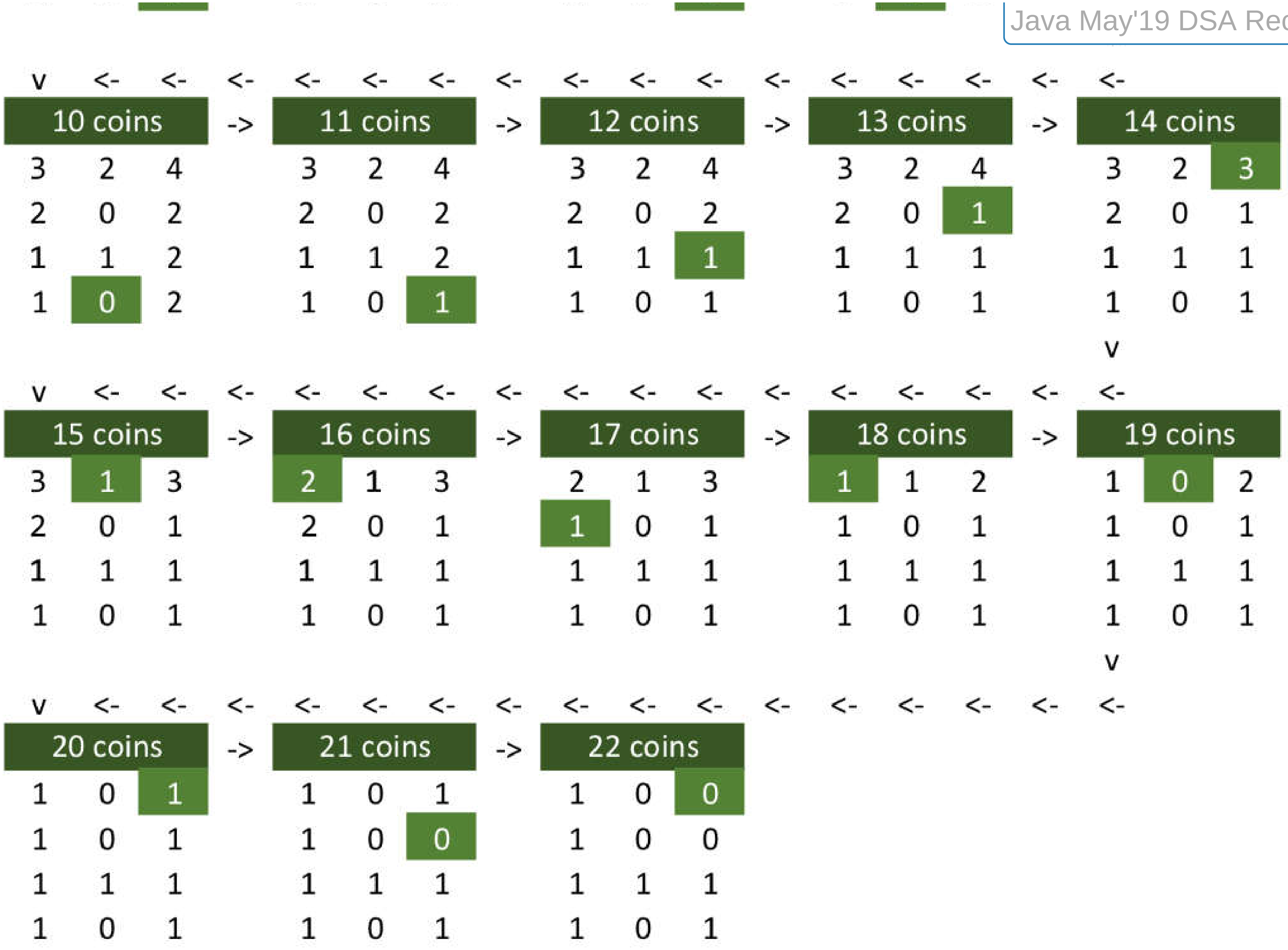
My submissions
All submissions
Best submissions

✓ **Points:** 100 (partial)
Time limit: 0.3s
JavaScript: 0.4s
Memory limit: 32M
JavaScript: 32M
Author: donchominkov

Tags
Arrays
Difficulty
Easy

Allowed languages
C#, java, JavaScript

Java May'19 DSA Recursion 2 - 1 day 21:40:06



```
v   <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-
  10 coins  ->    11 coins  ->    12 coins  ->    13 coins  ->    14 coins
3   2   4       3   2   4       3   2   4       3   2   4       3   2   3
2   0   2       2   0   2       2   0   2       2   0   1       2   0   1
1   1   2       1   1   2       1   1   1       1   1   1       1   1   1
1   0   2       1   0   1       1   0   1       1   0   1       1   0   1
                                                                v
v   <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-
  15 coins  ->    16 coins  ->    17 coins  ->    18 coins  ->    19 coins
3   1   3       2   1   3       2   1   3       1   1   2       1   0   2
2   0   1       2   0   1       1   0   1       1   0   1       1   0   1
1   1   1       1   1   1       1   1   1       1   1   1       1   1   1
1   0   1       1   0   1       1   0   1       1   0   1       1   0   1
                                                                v
v   <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-  <-
  20 coins  ->    21 coins  ->    22 coins
1   0   1       1   0   1       1   0   0
1   0   1       1   0   0       1   0   0
1   1   1       1   1   1       1   1   1
1   0   1       1   0   1       1   0   1
```

Scrooge McDuck is worried, not about his life, but if the coins he collect will be enough. Your task is to tell him how many coins he will collect, following the rules above.

- On the first line find **N** and **M**
  - The size of the labyrinth
- On the next **N lines** find **M integer values**, separated by a space
- The input data will always be valid and there is no need to check it explicitly
- The starting location of Scrooge McDuck will be marked as the only `0`

Output

- Print to the standard output
- On the single line, print the number of coins Scrooge McDuck can collect, following the rules

Contraints

- `2 <= N <= 10`
- `2 <= M <= 10`
- Each cell can contain up to 1024 coins

Sample tests

Input

Copy

```
4 3
3 2 4
2 0 3
1 1 5
2 2 5
```

Output

Copy

Input

Copy

```
3 3
10 10 0
10 10 10
10 10 10
```

Output

Copy

```
78
```

Input

Copy

```
3 3
10 10 10
10 0 10
10 10 10
```

Output

Copy

```
80
```

```
2 3
0 5 2
2 5 3
```

Output

Copy

```
15
```

Clarifications

on Dec. 22, 2017, 1:32 p.m.

Uploaded detailed example

on Dec. 22, 2017, 10:37 a.m.

Scrooge McDuck chooses left, right, up or down, only if the biggest value is in two or more neighbouring cells.

Telerik Academy | Powered by DMOJ