**CSCB324 Състезателно програмиране - I част**

**Проект – задача за олимпиада по програмиране**

**Пламен Костов F113851**

**Задача: "Магически лабиринт"**

**Описание:**

Даден е лабиринт с размери N × M, представен като матрица от символи. Лабиринтът съдържа:

* . — проходима клетка
* # — стена (непроходима клетка)
* S — начална позиция
* E — изход

Всяка клетка от лабиринта има *магическа стойност*, зададена от друга матрица със същите размери N × M, която съдържа цели числа. Целта е да се намери път от S до E, така че сумата от магическите стойности на клетките, през които преминаваш, да бъде **максимална**.

Можете да се движите само в четирите основни посоки: нагоре, надолу, наляво и надясно.

**Вход:**

1. Две цели числа N и M (1 ≤ N, M ≤ 50) — размерите на лабиринта.
2. N реда с по M символа — матрицата на лабиринта.
3. N реда с по M цели числа — матрицата на магическите стойности (1 ≤ цяло число ≤ 30).

**Изход:**

Един ред с едно число — максималната сума на магическите стойности по някой валиден път от S до E. Ако няма валиден път, изведете -1.

**Примерен вход 1:**

**A screenshot of a computer

Description automatically generated**

**Примерен изход 1:**

-1

**Обяснение:**

От началната позиция S (0,0) можем да се придвижим само към (1,0) или (0,1). Следвайки всички възможни пътища, виждаме, че няма как да достигнем до изхода E (3,5), тъй като всички маршрути към него са блокирани от стени (#). Затова програмата трябва да изведе -1, което означава, че няма валиден път!

**Примерен вход 2:**

A screenshot of a computer

Description automatically generated

**Примерен изход 2:**

50

**Обяснение:**

Отговорът се получава от 5 + 1 + 6 + 1 + 3 + 5 + 6 + 9 + 7 + 6 + 1 + 3 = 50. Има и други пътища като например 5 + 1 + 6 + 1 + 4 + 8 + 4 + 6 + 1 + 3 = 39, но в задачате се търси максиламната въжможна стойност!

**Problem: "Magical Maze"**

**Description:**

You are given a maze with dimensions N × M, represented as a matrix of characters. The maze contains:

* S — the starting position
* E — the exit
* . — passable cell
* # — wall (impassable cell)

Each cell in the maze has a *magical value* assigned to it, which is given by another matrix of the same size N × M, containing integers. Your task is to find a path from S to E such that the sum of the magical values of the cells you pass through is **maximum**.

You can move only in the four main directions: up, down, left, right.

**Input:**

1. Two integers N and M (1 ≤ N, M ≤ 50) — the dimensions of the maze.
2. N rows of M characters — the maze itself (composed of S, E, . and #).
3. N rows of M integers — the magical values of the maze cells (1 ≤ integer ≤ 30).

**Output:**

A single integer — the maximum sum of magical values along a valid path from S to E. If there is no valid path, output -1.

**Example Input 1:**

**A screenshot of a computer

Description automatically generated**

**Example Output 1:**

-1

**Explanation:**

From the starting position S (0,0) we can only move to (1,0) or (0,1). Following all possible paths, we see that we cannot reach the exit E (3,5), since all routes to it are blocked by walls (#). Therefore, the program should output -1, which means that there is no valid path!

**Example Input 2:**

A screenshot of a computer

Description automatically generated

**Example Output 2:**

50

**Explanation:**

The answer is obtained from 5 + 1 + 6 + 1 + 3 + 5 + 6 + 9 + 7 + 6 + 1 + 3 = 50. There are other ways, such as 5 + 1 + 6 + 1 + 4 + 8 + 4 + 6 + 1 + 3 = 39, but the problems seek the maximum possible value!