Mobile
Consulting Solutions

Assignment #1

Algorithms

Write a function that given an array of numbers (ie. [1, 2, 3, 4, 5]) perform a number of right circular rotations given by a specific index and return the resulting array. This means that every time that you perform a rotation, the last item on the array will be moved to the first position of that same array.

Example:

Array: [1, 2, 3, 4, 5]

Index: 3

Resulting Array [3, 4, 5, 1, 2]

Requirements:

- The solution must be encapsulated on a single function.
- The function must have input parameters and return the result correctly.

A mouse is placed in a maze. There is a chunk of cheese somewhere in the maze. The maze is represented as a n x m grid of integers where 0 represents a wall, 1 represents the path where the mouse can move and 9 represents the chunk of cheese. The mouse starts at the top left corner at (0,0).

Write an algorithm to determine whether the mouse can reach the chunk of cheese or not.

Example of 8x8 matrix where the mouse can get the cheese.

1 0 1 1 1 0 0 1

1 0 0 0 1 1 1 1

1 0 0 0 0 0 0 0

1 0 1 0 9 0 1 1

1 1 1 0 1 0 0 1

1 0 1 0 1 1 0 1

1 0 0 0 0 1 0 1

1 1 1 1 1 1 1 1

Input:

The input to the function/method consists of three arguments.

- grid - representing the two-dimensional integer grid.
- rows - an integer representing the number of rows in the grid (n).
- columns - an integer representing the number of columns in the grid (m).

Output:

Return an integer 1 if there is a path from the initial position of the mouse to the cheese, else return integer 0.

Note:

The mouse is not allowed to leave the grid or climb the walls.

Requirements:

- The solution must be written in Objective C
- The solution must be encapsulated on a single function.
- The function must have input parameters and return the result correctly.