Note: Rat in a Maze - Code Correction

Initialisation of solution 2D array

The code developed in the video may not work on your local IDE. This happens because the solution 2D array is not initialized with default value, before making recursive calls. Hence, to resolve it, please initialize the 2D-array, before initiating the recursive calls. The following code helps you understand where the initialization of solution 2D array has to be done:

```
void ratInAMaze(int maze[][20], int n){

  int** solution = new int*[n];
  for(int i=0;i<n;i++){
      solution[i] = new int[n];
  }
  // initialization of solution 2D arrays goes here.

  mazeHelp(maze,n,solution,0,0);

}
```

Complete code with initialisation of solution 2D array:

```
#include<bits/stdc++.h>
using namespace std;


void printSolution(int** solution,int n){
      for(int i=0;i<n;i++){
            for(int j=0;j<n;j++){
                  cout << solution[i][j] << " ";
            }
      }
      cout<<endl;
}
void mazeHelp(int maze[][20],int n,int** solution,int x,int y){
```

```
      if(x == n-1 && y == n-1){
            solution[x][y] =1;
            printSolution(solution,n);
            solution[x][y] =0;
            return;
      }
      if(x>=n || x<0 || y>=n || y<0 || maze[x][y] ==0 || solution[x][y]
==1){
            return;
      }
      solution[x][y] = 1;
      mazeHelp(maze,n,solution,x-1,y);
      mazeHelp(maze,n,solution,x+1,y);
      mazeHelp(maze,n,solution,x,y-1);
      mazeHelp(maze,n,solution,x,y+1);
      solution[x][y] = 0;
}
void ratInAMaze(int maze[][20], int n){

  int** solution = new int*[n];
  for(int i=0;i<n;i++){
      solution[i] = new int[n];
  }
  // Initialization of solution 2D array with 0
  for(int i=0; i<n; i++){
    memset(solution[i], 0, n*sizeof(int));
  }

  mazeHelp(maze,n,solution,0,0);

}
```