

Find the number of islands

```
class Graph:
    def __init__(self, row, col, g):
        self.ROW = row
        self.COL = col
        self.graph = g
    def isSafe(self, i, j, visited):
        return (i >= 0 and i < self.ROW and
                j >= 0 and j < self.COL and
                not visited[i][j] and self.graph[i][j])
    def DFS(self, i, j, visited):
        rowNbr = [-1, -1, -1, 0, 0, 1, 1, 1]
        colNbr = [-1, 0, 1, -1, 1, -1, 0, 1]
        visited[i][j] = True
        for k in range(8):
            if self.isSafe(i + rowNbr[k], j + colNbr[k], visited):
                self.DFS(i + rowNbr[k], j + colNbr[k], visited)
    def countIslands(self):
        visited = [[False for j in range(self.COL)] for i in range(self.ROW)]
        count = 0
        for i in range(self.ROW):
            for j in range(self.COL):
                if visited[i][j] == False and self.graph[i][j] == 1:
                    self.DFS(i, j, visited)
                    count += 1
        return count

graph = [[1, 1, 0, 0, 0],
         [0, 1, 0, 0, 1],
         [1, 0, 0, 1, 1],
         [0, 0, 0, 0, 0],
         [1, 0, 1, 0, 1]]

row = len(graph)
col = len(graph[0])
g = Graph(row, col, graph)
print("Number of islands is:")
print(g.countIslands())
```