

Problem – 8 Given a boolean 2D matrix grid of size N x M. You have to find the number of distinct islands where a group of connected 1s (horizontally or vertically) forms an island. Two islands are considered to be distinct if and only if one island is equal to another (not rotated or reflected).

```
def numDistinctIslands(grid):
```

```
    def dfs(x, y, shape):
```

```
        if 0 <= x < len(grid) and 0 <= y < len(grid[0]) and grid[x][y] == 1:
```

```
            grid[x][y] = 0
```

```
            shape.append((x, y))
```

```
            dfs(x + 1, y, shape)
```

```
            dfs(x - 1, y, shape)
```

```
            dfs(x, y + 1, shape)
```

```
            dfs(x, y - 1, shape)
```

```
    unique_shapes = set()
```

```
    for i in range(len(grid)):
```

```
        for j in range(len(grid[0])):
```

```
            if grid[i][j] == 1:
```

```
                shape = []
```

```
                dfs(i, j, shape)
```

```
                unique_shapes.add(tuple(shape))
```

```
    return len(unique_shapes)
```

```
grid = [
```

```
    [1, 1, 0, 0, 0],
```

```
    [1, 1, 0, 0, 0],
```

```
    [0, 0, 0, 1, 1],
```

```
    [0, 0, 0, 1, 1]
```

```
]
```

```
print(numDistinctIslands(grid))
```

