

Quiz 2

Name: Schaquille Devlin Aristano
NRP: 5025211211

For Quiz 2, I am making a random maze generator using randomized DFS algorithm (recursive backtracking).

Design

Maze generation is done by using a 2d array to create a grid of width*height size and using a stack to keep track of previously visited cells.

We would randomly traverse the grid until we arrive in a cell with no unvisited neighbouring cell which we will then backtrack using the previously mentioned stack(thus the name) until a cell with an unvisited neighbour that will allow us to continue our random traversal.

This process repeats until all cell have been visited which can be tracked using a simple integer variable.

Implementation

(incomplete)

```
#include <iostream>
#include <stack>
#include <vector>
#include <cstdlib>
#include <ctime>

using namespace std;

enum STATE{north=1, east=2, south=4, west=8, visited=16};

int main()
{
    int width = 40;
    int height = 25;
    int visited, curr_x, curr_y;
    int maze[width][height];
    stack<pair<int, int>> visitedCell;

    //init
    srand(time(NULL));

    curr_x = curr_y = 0;
    visitedCell.push(make_pair(0,0));
    visited = 1;

    vector<int> neighbour;
```

```
    while (visited < width*height)
    {
        if (visitedCell.top().second > 0 && (maze[curr_x][curr_y-1] == 16))
//North neighbour
        {

        }
    }

    return 0;
}
```

Analysis