

Moves in Best Path

In this program we will find out how many moves there are along the best path from spot A to spot B. We can only move North, South, East or West

```
import java.util.Scanner;
import java.io.*;

public class matrixsearch
{
    private int[][] mat;

    public matrixsearch()
    {
        mat = new int[][]{
            { 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0 },
            { 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0 },
            { 0, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 1, 1, 0, 0 },
            { 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0 },
            { 0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 7, 0, 0, 0, 0 }
        };
    }

    // find out how many moves it is to the best path
    // you are going from starting position 0,0 to where there is a 7 in the matrix
    // 1 means there is a path - 0 is a block

    public int bestPath(int r,int c)
    {
        }
    }
}
```

Client

```
public void main()
{int b = bestPath(0,0);
  System.out.println("Best path is "+b);
}
```