ShortestPathBinaryMatrix.java

```
1    package com.example.graph;
2
3    import java.util.Arrays;
4    import java.util.LinkedList;
5    import java.util.Queue;
6
7    class tuple {
8        int a;
9        int b;
10       int c;
11
12       tuple(int _a, int _b, int _c) {
13           this.a = _a;
14           this.b = _b;
15           this.c = _c;
16       }
17   }
18
19   public class ShortestPathBinaryMatrix {
20       public int shortestPathBinaryMatrix(int[][] grid) {
21           int n = grid.length;
22           int m = grid[0].length;
23 4         if (grid[0][0] != 0 || grid[n - 1][n - 1] != 0)
24 1             return -1;
25 2         if (n - 1 == 0)
26 1             return 1;
27           int[][] dis = new int[n][m];
28           Queue<tuple> queue = new LinkedList<>();
29           for (int[] i : dis) {
30 1             Arrays.fill(i, (int) 1e9);
31           }
32           dis[0][0] = 1;
33           queue.add(new tuple(1, 0, 0));
34           int[] dr = { 0, 1, -1, 0, 1, -1, 1, -1 };
35           int[] dc = { 1, 0, 0, -1, 1, 1, -1, -1 };
36 1         while (!queue.isEmpty()) {
37               tuple it = queue.poll();
38               int d = it.a;
39               int r = it.b;
40               int c = it.c;
41 2             for (int i = 0; i < 8; i++) {
42 1                 int nrow = r + dr[i];
43 1                 int ncol = c + dc[i];
44 12                if (nrow >= 0 && nrow < n && ncol >= 0 && ncol < m && grid[nrow][ncol] == 0
45                                && d + 1 < dis[nrow][ncol]) {
46 1                     dis[nrow][ncol] = 1 + d;
47 4                     if (nrow == n - 1 && ncol == n - 1)
48 2                         return d + 1;
49 1                     queue.add(new tuple(d + 1, nrow, ncol));
50                  }
51              }
52          }
53 1       return -1;
54       }
55   }
```

Mutations

23
1. negated conditional → KILLED
2. Replaced integer subtraction with addition → KILLED
3. Replaced integer subtraction with addition → KILLED
4. negated conditional → KILLED

24
1. replaced int return with 0 for com/example/graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → KILLED

25
1. Replaced integer subtraction with addition → KILLED
2. negated conditional → KILLED

26	1. replaced int return with 0 for com/example/graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → KILLED
30	1. removed call to java/util/Arrays::fill → KILLED
36	1. negated conditional → KILLED
41	1. changed conditional boundary → KILLED 2. negated conditional → KILLED
42	1. Replaced integer addition with subtraction → SURVIVED
43	1. Replaced integer addition with subtraction → SURVIVED
44	1. negated conditional → KILLED 2. negated conditional → KILLED 3. negated conditional → KILLED 4. Replaced integer addition with subtraction → SURVIVED 5. changed conditional boundary → KILLED 6. negated conditional → KILLED 7. changed conditional boundary → KILLED 8. negated conditional → KILLED 9. changed conditional boundary → SURVIVED 10. changed conditional boundary → SURVIVED 11. changed conditional boundary → SURVIVED 12. negated conditional → KILLED
46	1. Replaced integer addition with subtraction → SURVIVED
47	1. negated conditional → KILLED 2. negated conditional → KILLED 3. Replaced integer subtraction with addition → KILLED 4. Replaced integer subtraction with addition → KILLED
48	1. Replaced integer addition with subtraction → KILLED 2. replaced int return with 0 for com/example/graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → KILLED
49	1. Replaced integer addition with subtraction → KILLED
53	1. replaced int return with 0 for com/example/graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → NO_COVERAGE

Active mutators

- CONDITIONALS_BOUNDARY
- EMPTY_RETURNS
- FALSE_RETURNS
- INCREMENTS
- INVERT_NEGS
- MATH
- NEGATE_CONDITIONALS
- NULL_RETURNS
- PRIMITIVE_RETURNS
- TRUE_RETURNS
- VOID_METHOD_CALLS

Tests examined

- com.example.graph.ShortestPathBinaryMatrixTest.testShortestPathBinaryMatrix(com.example.graph.ShortestPathBinaryMatrixTest) (0 ms)

Report generated by PIT 1.15.0