ShortestPathBinaryMatrix.java

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

Mutations

```
21   1. Replaced integer subtraction with addition → KILLED
     2. negated conditional → KILLED
     3. negated conditional → KILLED
     4. Replaced integer subtraction with addition → KILLED
     5. replaced int return with 0 for Graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → KILLED
22   1. replaced int return with 0 for Graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → NO_COVERAGE
     2. Replaced integer subtraction with addition → SURVIVED
     3. negated conditional → KILLED
26   1. removed call to java/util/Arrays::fill → KILLED
32   1. negated conditional → KILLED
37   1. changed conditional boundary → KILLED
     2. negated conditional → KILLED
38   1. Replaced integer addition with subtraction → SURVIVED
39   1. Replaced integer addition with subtraction → SURVIVED
40   1. negated conditional → KILLED
     2. negated conditional → KILLED
     3. changed conditional boundary → SURVIVED
     4. changed conditional boundary → SURVIVED
     5. negated conditional → KILLED
     6. negated conditional → KILLED
     7. changed conditional boundary → SURVIVED
     8. Replaced integer addition with subtraction → SURVIVED
     9. changed conditional boundary → KILLED
     10. changed conditional boundary → KILLED
     11. negated conditional → KILLED
     12. negated conditional → KILLED
41   1. Replaced integer addition with subtraction → SURVIVED
42   1. Replaced integer subtraction with addition → KILLED
     2. negated conditional → KILLED
     3. Replaced integer subtraction with addition → KILLED
     4. Replaced integer addition with subtraction → KILLED
```

```
      5. negated conditional → KILLED
      6. replaced int return with 0 for Graph/ShortestPathBinaryMatrix::shortestPathBinaryMatrix → KILLED
43    1. Replaced integer addition with subtraction → KILLED
47    1. replaced int return with 0 for 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

- Graph.ShortestPathBinaryMatrixTest.testShortestPathBinaryMatrix(Graph.ShortestPathBinaryMatrixTest) (0 ms)

Report generated by PIT 1.15.0