NumberofDistinctSubstring.java

```
1   package com.example.trie;
2
3   public class NumberofDistinctSubstring {
4
5       public int countDistinctSubstrings(String s) {
6           Node root = new Node();
7           int n = s.length();
8           int count = 0;
9
10 2        for (int i = 0; i < n; i++) {
11              Node node = root;
12
13 2            for (int j = i; j < n; j++) {
14 1                if (!node.containsKey(s.charAt(j))) {
15 1                    node.put(s.charAt(j), new Node());
16 1                    count++;
17                  }
18                  node = node.get(s.charAt(j));
19              }
20          }
21 2        return count + 1;
22      }
23  }
```

Mutations

10	1. negated conditional → KILLED 2. changed conditional boundary → SURVIVED
13	1. negated conditional → KILLED 2. changed conditional boundary → KILLED
14	1. negated conditional → KILLED
15	1. removed call to com/example/trie/Node::put → KILLED
16	1. Changed increment from 1 to -1 → KILLED
21	1. Replaced integer addition with subtraction → KILLED 2. replaced int return with 0 for com/example/trie/NumberofDistinctSubstring::countDistinctSubstrings → KILLED

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.trie.NumberofDistinctSubstringTest.main(com.example.trie.NumberofDistinctSubstringTest) (0 ms)

Report generated by PIT 1.15.0