NumeroPerfeito.java

```
1   package com.example.mutationspringtest.methods;
2
3   public class NumeroPerfeito {
4       public static boolean find(int n)
5       {
6           int soma = 0;
7  2        if(n < 1)
8           {
9  1            return false;
10          }
11 3        for (int i = n; i > 0; i--)
12          {
13 3            if (i != n && n % i ==0 )
14              {
15 1                soma += i;
16              }
17          }
18 1        if (soma == n)
19          {
20 1            return true;
21          }
22          else
23          {
24 1            return false;
25          }
26      }
27  }
```

Mutations

7
1. changed conditional boundary → NO_COVERAGE
2. negated conditional → NO_COVERAGE

9
1. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE

11
1. changed conditional boundary → NO_COVERAGE
2. Changed increment from -1 to 1 → NO_COVERAGE
3. negated conditional → NO_COVERAGE

13
1. Replaced integer modulus with multiplication → NO_COVERAGE
2. negated conditional → NO_COVERAGE
3. negated conditional → NO_COVERAGE

15
1. Replaced integer addition with subtraction → NO_COVERAGE

18
1. negated conditional → NO_COVERAGE

20
1. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE

24
1. replaced return of integer sized value with (x == 0 ? 1 : 0) → NO_COVERAGE

Active mutators

- CONDITIONALS_BOUNDARY_MUTATOR
- INCREMENTS_MUTATOR
- INVERT_NEGS_MUTATOR
- MATH_MUTATOR
- NEGATE_CONDITIONALS_MUTATOR
- RETURN_VALS_MUTATOR
- VOID_METHOD_CALL_MUTATOR

Tests examined

Report generated by PIT 1.4.10