









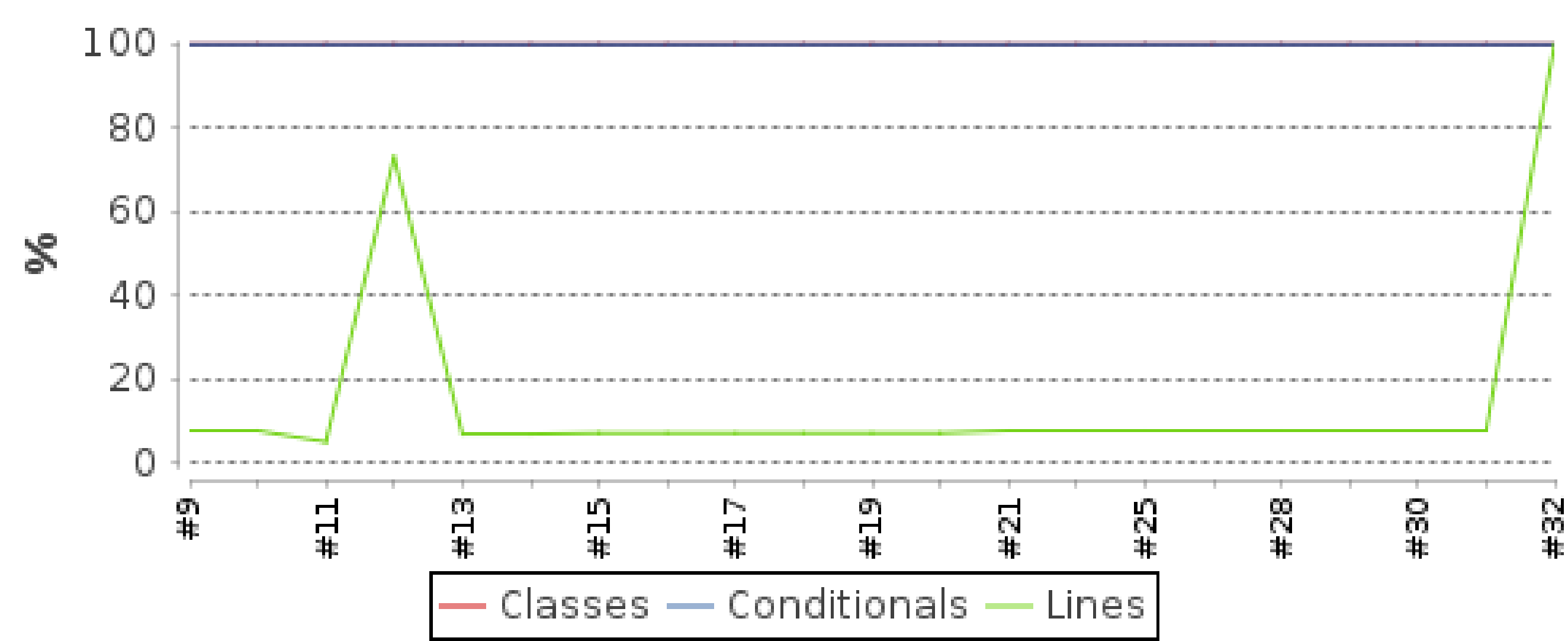
-  Status
-  Changes
-  Console Output
-  Edit Build Information
-  Delete build '#32'
-  Git Build Data
-  Coverage Report
-  Previous Build

Code Coverage

[Cobertura Coverage Report](#) > [lab07](#) >

test_calculator.py

Trend



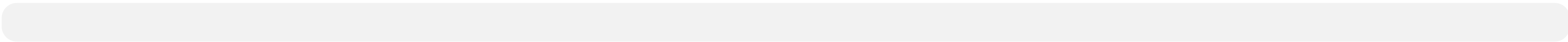
File Coverage summary

Name	Classes	Lines	Conditionals
test_calculator.py	100% <div>1/1</div>	100% <div>26/26</div>	100% <div>0/0</div>

Coverage Breakdown by Class

Name	Lines	Conditionals
test_calculator.py	100% <div>26/26</div>	N/A

Source



lab07/test_calculator.py

```
1 1 import unittest
2 1 import calculator
3
4
5 1 class TestCalculator(unittest.TestCase):
6 1     def setUp(self):
7 1         self.calculator = calculator.Calculator()
8
9 1     def test_add(self):
10 1         self.assertEqual(self.calculator.add(3, 4), 7)
11 1         self.assertEqual(self.calculator.add(-1, 1), 0)
12 1         self.assertEqual(self.calculator.add(0, 0), 0)
13
14 1     def test_subtract(self):
15 1         self.assertEqual(self.calculator.subtract(5, 3), 2)
16 1         self.assertEqual(self.calculator.subtract(-1, 1), -2)
17 1         self.assertEqual(self.calculator.subtract(0, 0), 0)
18
19 1     def test_multiply(self):
20 1         self.assertEqual(self.calculator.multiply(2, 3), 6)
21 1         self.assertEqual(self.calculator.multiply(-1, 5), -5)
22 1         self.assertEqual(self.calculator.multiply(0, 0), 0)
23
24 1     def test_divide(self):
25 1         self.assertEqual(self.calculator.divide(8, 2), 4)
26 1         self.assertEqual(self.calculator.divide(-6, 3), -2)
27 1         self.assertEqual(self.calculator.divide(0, 5), 0)
28 1         with self.assertRaises(ValueError):
29 1             self.calculator.divide(5, 0)
30
31
32 1 if __name__ == '__main__':
33 1     unittest.main()
```