Software Engineering – Lab 09

Mutation Testing

Tanay Kewalramani

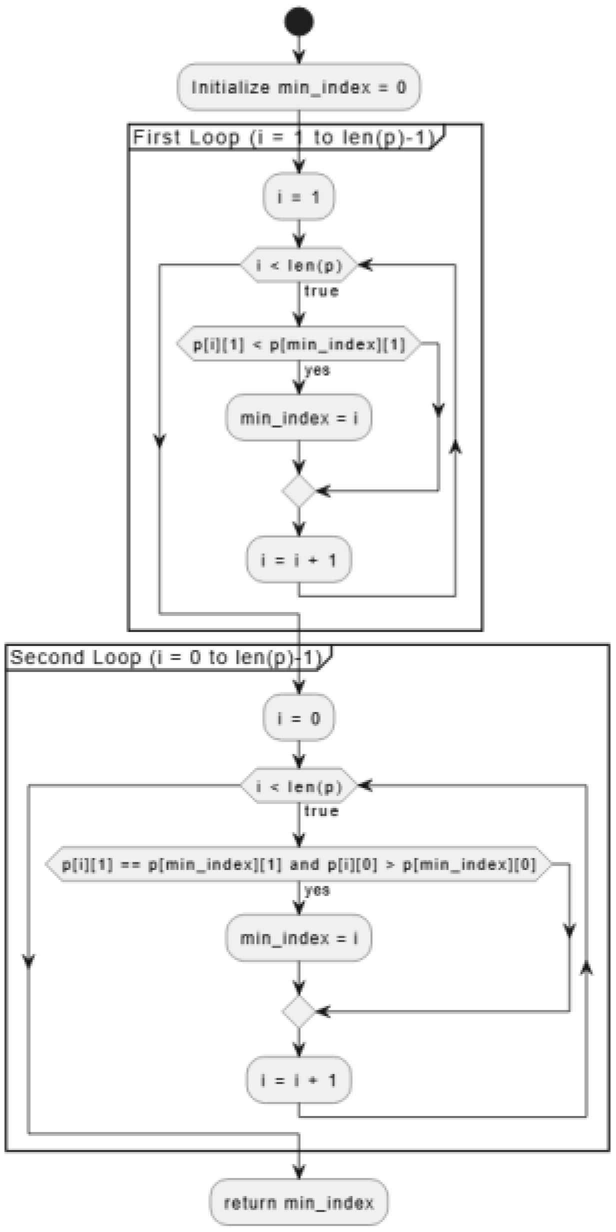
202201362

Problem 1:

1. doGraham Function () Converting code to Python:

```
def do_graham(p):
    min_index = 0

    # Search for minimum point
    for i in range(1, len(p)):
        if p[i][1] < p[min_index][1]: # Compare y values
            min_index = i

    # Continue along the values with the same y component
    for i in range(len(p)):
        if p[i][1] == p[min_index][1] and p[i][0] > p[min_index][0]: # Compare x
values
            min_index = i

    return min_index # Return the index of the minimum point
```

<u>Corresponding ontrol flow graph:</u>



Initialize min_index = 0
First Loop (i = 1 to len(p)-1)
i = 1
i < len(p)
true
p[i][1] < p[min_index][1]
yes
min_index = i
i = i + 1
Second Loop (i = 0 to len(p)-1)
i = 0
i < len(p)
true
p[i][1] == p[min_index][1] and p[i][0] > p[min_index][0]
yes
min_index = i
i = i + 1
return min_index

The logic in the control flow graph matches the one generated by Eclipse's flow graph generator.

2. Test sets:

ii. Test Sets

a. Statement Coverage

This means running the code so that every line is used at least once.

- Examples to test this:
  - `` `p = [(1, 2), (2, 3), (0, 1)]` `` ➔ (Minimum point: (0, 1), index 2)
  - `` `p = [(1, 2), (2, 1), (0, 1)]` `` ➔ (Minimum point: (0, 1), index 2)
  - `` `p = [(1, 2), (1, 2), (0, 1)]` `` ➔ (Minimum point: (0, 1), index 2)

b. Branch Coverage

This makes sure each condition in the code is checked for both true and false outcomes.

- Examples to test this:
  - `` `p = [(1, 2), (2, 3), (0, 1)]` `` ➔ (First condition True)
  - `` `p = [(1, 2), (1, 3), (0, 1)]` `` ➔ (First condition False)
  - `` `p = [(1, 1), (2, 1), (0, 1)]` `` ➔ (Second condition True)
  - `` `p = [(1, 1), (0, 1), (0, 2)]` `` ➔ (Second condition False)

c. Basic Condition Coverage

This means testing each simple (or “atomic”) part of the code to make sure it’s both true and false.

- Examples (same as above):

- ` p = [(1, 2), (2, 3), (0, 1)]` ➔ (First condition True)

- ` p = [(1, 2), (1, 3), (0, 1)]` ➔ (First condition False)

- ` p = [(1, 1), (2, 1), (0, 1)]` ➔ (Second condition True)

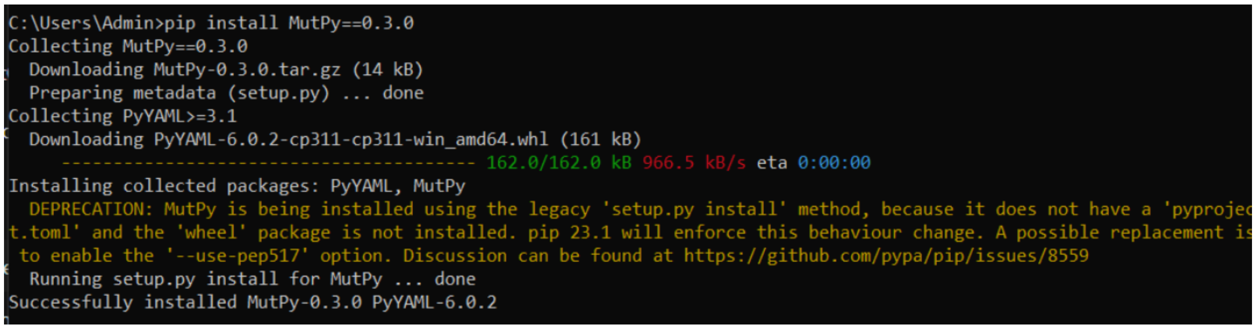
- ` p = [(1, 1), (0, 1), (0, 2)]` ➔ (Second condition False)

iii. Mutation Testing

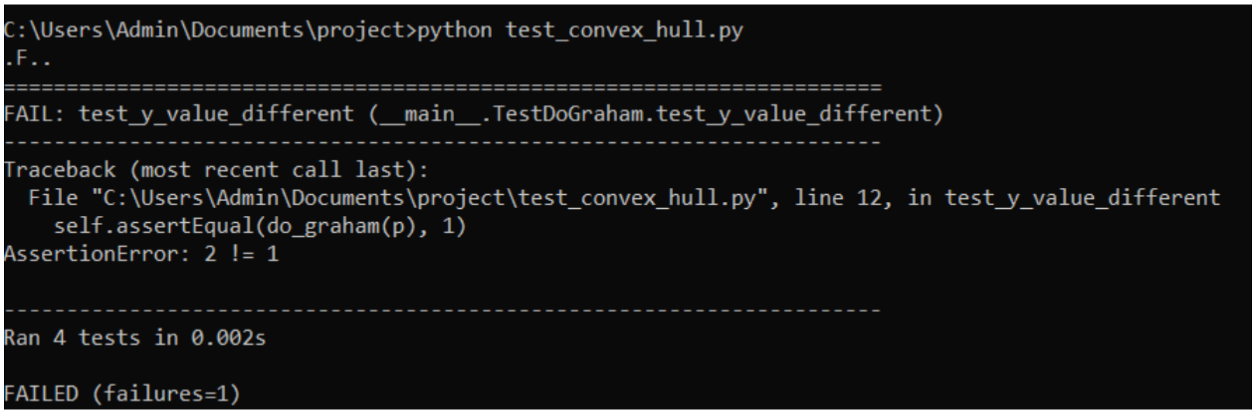
This is about testing if slight changes (mutations) in the code can be caught by the tests. Types of mutations:

- Removing a line
- Adding a new line
- Changing a line

MutPy testing:



```
C:\Users\Admin>pip install MutPy==0.3.0
Collecting MutPy==0.3.0
  Downloading MutPy-0.3.0.tar.gz (14 kB)
  Preparing metadata (setup.py) ... done
Collecting PyYAML>=3.1
  Downloading PyYAML-6.0.2-cp311-cp311-win_amd64.whl (161 kB)
     ---------------------------------------- 162.0/162.0 kB 966.5 kB/s eta 0:00:00
Installing collected packages: PyYAML, MutPy
  DEPRECATION: MutPy is being installed using the legacy 'setup.py install' method, because it does not have a 'pyprojec
t.toml' and the 'wheel' package is not installed. pip 23.1 will enforce this behaviour change. A possible replacement is
 to enable the '--use-pep517' option. Discussion can be found at https://github.com/pypa/pip/issues/8559
  Running setup.py install for MutPy ... done
Successfully installed MutPy-0.3.0 PyYAML-6.0.2
```



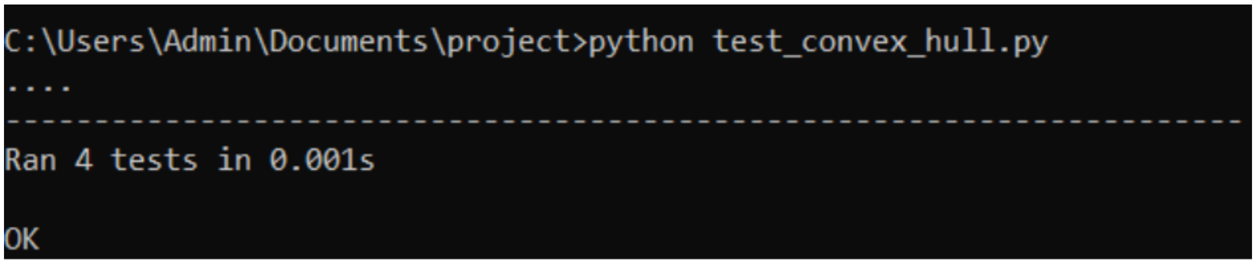
```
C:\Users\Admin\Documents\project>python test_convex_hull.py
.F..
======================================================================
FAIL: test_y_value_different (__main__.TestDoGraham.test_y_value_different)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 12, in test_y_value_different
    self.assertEqual(do_graham(p), 1)
AssertionError: 2 != 1

----------------------------------------------------------------------
Ran 4 tests in 0.002s

FAILED (failures=1)
```

Code corrections:

```
def do_graham(p):
    min_index = 0

    for i in range(len(p)):
        if p[i][1] < p[min_index][1]:
            min_index = i
        elif p[i][1] == p[min_index][1] and p[i][0] < p[min_index][0]:
            min_index = i

    return min_index
```

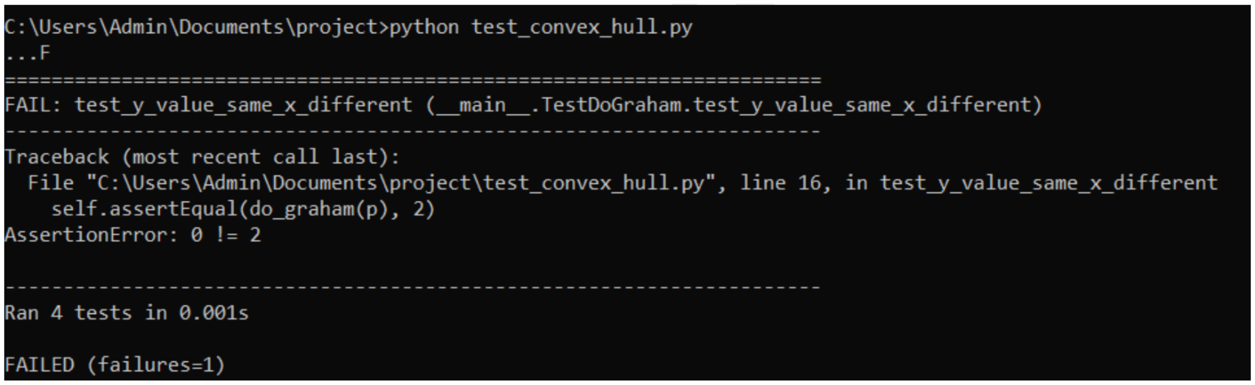
Test cases:

```
import unittest
from convex_hull import do_graham

class TestDoGraham(unittest.TestCase):
    def test_basic_case(self):
        p = [[0, 0], [1, 1]]
        self.assertEqual(do_graham(p), 0)
    def test_y_value_different(self):
        p = [[0, 0], [1, -1]]
        self.assertEqual(do_graham(p), 1)

    def test_y_value_same_x_different(self):
        p = [[0, 0], [0, 0], [1, 0]]
        self.assertEqual(do_graham(p), 2)

    def test_y_value_edge_case(self):
        p = [[1, 1], [1, 1], [2, 0]]
        self.assertEqual(do_graham(p), 2)

if __name__ == 'main':
    unittest.main()
```



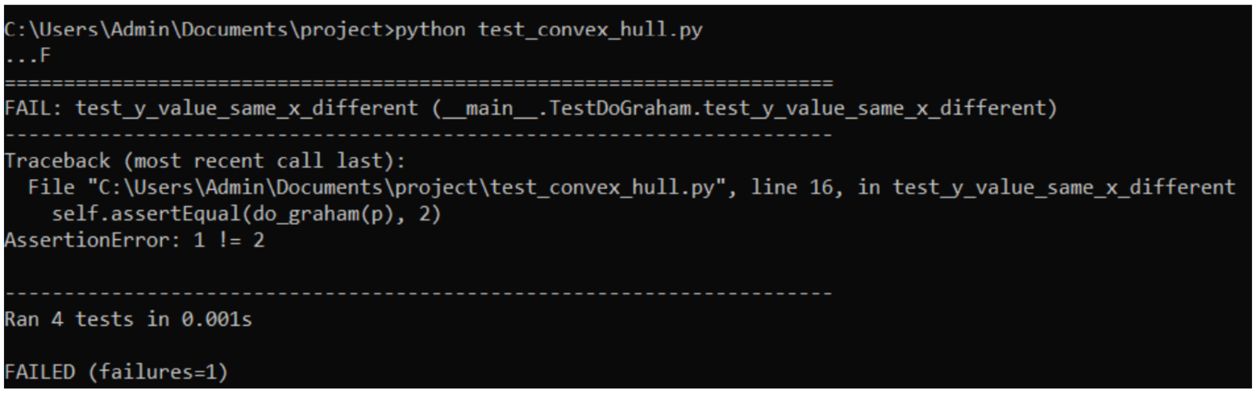
```
C:\Users\Admin\Documents\project>python test_convex_hull.py
....
----------------------------------------------------------------------
Ran 4 tests in 0.001s

OK
```

```
# Mutation 1: Deleting a line of code
def do_graham(p):
    min_index = 0
    for i in range(len(p)):
        if p[i][1] < p[min_index][1]:
            min_index = i


    return min_index
```



```
C:\Users\Admin\Documents\project>python test_convex_hull.py
...F
======================================================================
FAIL: test_y_value_same_x_different (__main__.TestDoGraham.test_y_value_same_x_different)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 16, in test_y_value_same_x_different
    self.assertEqual(do_graham(p), 2)
AssertionError: 0 != 2

----------------------------------------------------------------------
Ran 4 tests in 0.001s

FAILED (failures=1)
```

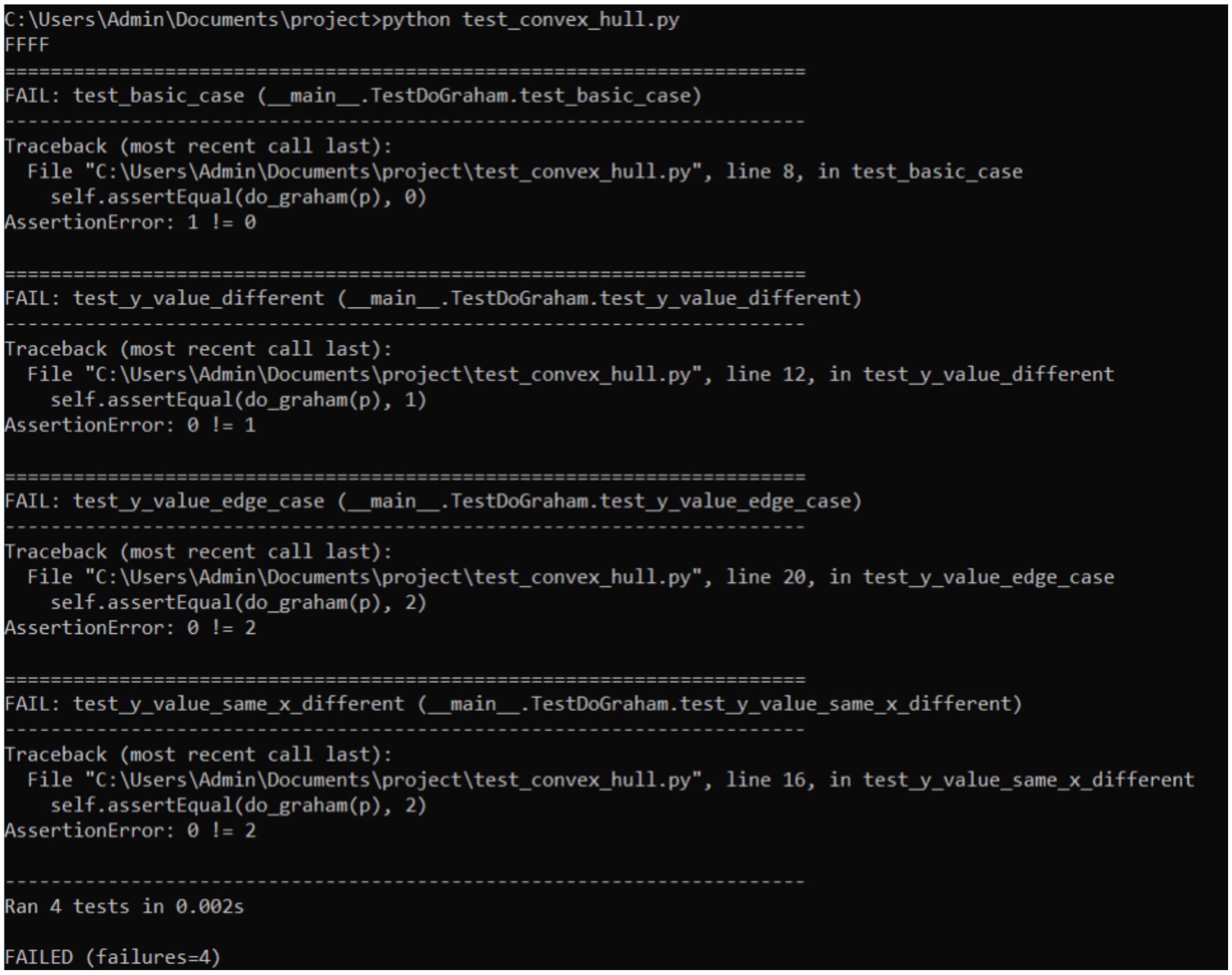
```
# Mutation 2: Inserting a line of code
def do_graham(p):
    min_index = 0
    min_index = 1


    for i in range(len(p)):
        if p[i][1] < p[min_index][1]:
            min_index = i
        elif p[i][1] == p[min_index][1] and p[i][0] < p[min_index][0]:
            min_index = i


    return min_index
```



```
C:\Users\Admin\Documents\project>python test_convex_hull.py
...F
======================================================================
FAIL: test_y_value_same_x_different (__main__.TestDoGraham.test_y_value_same_x_different)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 16, in test_y_value_same_x_different
    self.assertEqual(do_graham(p), 2)
AssertionError: 1 != 2

----------------------------------------------------------------------
Ran 4 tests in 0.001s

FAILED (failures=1)
```

Mutation 3:

```
def do_graham(p):
    min_index = 0

    for i in range(len(p)):
        if p[i][1] > p[min_index][1]:
            min_index = i
        elif p[i][1] == p[min_index][1] and p[i][0] < p[min_index][0]:
            min_index = i

    return min_index
```



```
C:\Users\Admin\Documents\project>python test_convex_hull.py
FFFF
======================================================================
FAIL: test_basic_case (__main__.TestDoGraham.test_basic_case)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 8, in test_basic_case
    self.assertEqual(do_graham(p), 0)
AssertionError: 1 != 0

======================================================================
FAIL: test_y_value_different (__main__.TestDoGraham.test_y_value_different)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 12, in test_y_value_different
    self.assertEqual(do_graham(p), 1)
AssertionError: 0 != 1

======================================================================
FAIL: test_y_value_edge_case (__main__.TestDoGraham.test_y_value_edge_case)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 20, in test_y_value_edge_case
    self.assertEqual(do_graham(p), 2)
AssertionError: 0 != 2

======================================================================
FAIL: test_y_value_same_x_different (__main__.TestDoGraham.test_y_value_same_x_different)
----------------------------------------------------------------------
Traceback (most recent call last):
  File "C:\Users\Admin\Documents\project\test_convex_hull.py", line 16, in test_y_value_same_x_different
    self.assertEqual(do_graham(p), 2)
AssertionError: 0 != 2

----------------------------------------------------------------------
Ran 4 tests in 0.002s

FAILED (failures=4)
```

iv. Path Coverage

Here, we test each loop in the code to make sure it's run 0, 1, or 2 times.

- Examples for path coverage:
  - `p = []` ➔ (No loops are run)
  - `p = [(1, 1)]` ➔ (First loop runs 0 times, second loop 0 times)
  - `p = [(1, 1), (1, 1), (1, 1)]` ➔ (First loop runs 1 time, second loop 0 times)
  - `p = [(1, 2), (2, 1), (0, 1)]` ➔ (Both loops run 1 time)

- ` p = [(1, 2), (0, 1), (0, 1)]` ➔ (Second loop runs 2 times)