MutPy mutation report

04.11.2024 16:55

Target

- `main_code.py`

Tests [6]

- `test_code` [0.0 s]

Result summary

- **Score** - 75.0%
- **Time** - 0.2 s

Mutants [8]

- killed - 6
- survived - 2
- incompetent - 0
- timeout - 0

75.0% | 25.0%

#	Module	Operator	Tests run	Duration	Result
1		COI [11]	1	0.008 s	killed
2		COI [15]	1	0.008 s	killed
3		LCR [15]	1	0.008 s	killed
4		ROR [11]	1	0.008 s	killed
5		ROR [11]	6	0.006 s	survived
6		ROR [15]	1	0.008 s	killed
7		ROR [15]	2	0.009 s	killed
8		ROR [15]	6	0.007 s	survived

Mutation #1

Details

- module -
- killed by `test_basic_condition_coverage_1 (test_code.TestDoGraham)`
- duration - 0.008 s
- tests run - 1

Exception traceback

```
Traceback (most recent call last):
  File "/content/test_code.py", line 29, in test_basic_condition_coverage_1
    self.assertEqual((result.x, result.y), expected, f"Failed for points: {[(p.x, p.y) for p in points]}")
AssertionError: Tuples differ: (0, 1) != (1, -1)

First differing element 0:
0
1

- (0, 1)
?  ^

+ (1, -1)
?  ^  +
 : Failed for points: [(0, 1), (1, -1), (2, 0)]
```

Mutations

- COI - line 11

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if not (points[i].y < points[min_index].y):
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y == points[min_index].y and points[i].x > points[min_index].x):
            min_index = i

    return points[min_index]
```

Mutation #2

Details

- module -
- killed by `test_basic_condition_coverage_1 (test_code.TestDoGraham)`
- duration - 0.008 s
- tests run - 1

Exception traceback

```
Traceback (most recent call last):
  File "/content/test_code.py", line 29, in test_basic_condition_coverage_1
    self.assertEqual((result.x, result.y), expected, f"Failed for points: {[(p.x, p.y) for p in points]}")
AssertionError: Tuples differ: (2, 0) != (1, -1)

First differing element 0:
2
1

- (2, 0)
+ (1, -1) : Failed for points: [(0, 1), (1, -1), (2, 0)]
```

Mutations

- COI - line 15

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y < points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if not ((points[i].y == points[min_index].y and points[i].x > points[min_index].x)):
            min_index = i

    return points[min_index]
```

Mutation #3

Details

- module -
- killed by `test_basic_condition_coverage_1 (test_code.TestDoGraham)`
- duration - 0.008 s
- tests run - 1

Exception traceback

```
Traceback (most recent call last):
  File "/content/test_code.py", line 29, in test_basic_condition_coverage_1
    self.assertEqual((result.x, result.y), expected, f"Failed for points: {[(p.x, p.y) for p in points]}")
AssertionError: Tuples differ: (2, 0) != (1, -1)

First differing element 0:
2
1

- (2, 0)
+ (1, -1) : Failed for points: [(0, 1), (1, -1), (2, 0)]
```

Mutations

- LCR - line 15

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y < points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y == points[min_index].y or points[i].x > points[min_index].x):
            min_index = i

    return points[min_index]
```

Mutation #4

Details

- module -
- killed by `test_basic_condition_coverage_1 (test_code.TestDoGraham)`
- duration - 0.008 s
- tests run - 1

Exception traceback

```
Traceback (most recent call last):
  File "/content/test_code.py", line 29, in test_basic_condition_coverage_1
    self.assertEqual((result.x, result.y), expected, f"Failed for points: {[(p.x, p.y) for p in points]}")
AssertionError: Tuples differ: (0, 1) != (1, -1)

First differing element 0:
0
1

- (0, 1)
?  ^

+ (1, -1)
?  ^  +
 : Failed for points: [(0, 1), (1, -1), (2, 0)]
```

Mutations

- ROR - line 11

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y > points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y == points[min_index].y and points[i].x > points[min_index].x):
            min_index = i

    return points[min_index]
```

Mutation #5

Details

- module -
- survived
- duration - 0.006 s
- tests run - 6

Mutations

- ROR - line 11

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y <= points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y == points[min_index].y and points[i].x > points[min_index].x):
            min_index = i

    return points[min_index]
```

Mutation #6

Details

- module -
- killed by `test_basic_condition_coverage_1 (test_code.TestDoGraham)`
- duration - 0.008 s
- tests run - 1

Exception traceback

```
Traceback (most recent call last):
  File "/content/test_code.py", line 29, in test_basic_condition_coverage_1
    self.assertEqual((result.x, result.y), expected, f"Failed for points: {[(p.x, p.y) for p in points]}")
AssertionError: Tuples differ: (2, 0) != (1, -1)

First differing element 0:
2
1

- (2, 0)
+ (1, -1) : Failed for points: [(0, 1), (1, -1), (2, 0)]
```

Mutations

- ROR - line 15

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y < points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y != points[min_index].y and points[i].x > points[min_index].x):
            min_index = i

    return points[min_index]
```

Mutation #7

Details

- module -
- killed by `test_basic_condition_coverage_2 (test_code.TestDoGraham)`
- duration - 0.009 s
- tests run - 2

Exception traceback

```
Traceback (most recent call last):
  File "/content/test_code.py", line 35, in test_basic_condition_coverage_2
    self.assertEqual((result.x, result.y), expected, f"Failed for points: {[(p.x, p.y) for p in points]}")
AssertionError: Tuples differ: (0, 0) != (1, 0)

First differing element 0:
0
1

- (0, 0)
?  ^

+ (1, 0)
?  ^
 : Failed for points: [(0, 0), (1, 0), (2, 1)]
```

Mutations

- ROR - line 15

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y < points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y == points[min_index].y and points[i].x < points[min_index].x):
            min_index = i

    return points[min_index]
```

Mutation #8

Details

- module -
- survived
- duration - 0.007 s
- tests run - 6

Mutations

- ROR - line 15

Mutant

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

def doGraham(points):
    min_index = 0

    for i in range(1, len(points)):
        if points[i].y < points[min_index].y:
            min_index = i

    for i in range(0, len(points)):
        if (points[i].y == points[min_index].y and points[i].x >= points[min_index].x):
            min_index = i

    return points[min_index]
```