Software Testing HW4 - Mutation Testing

(a) How many mutants are there in total?

- According to the report generated by StrykerJS, there are 106 mutants in total.

(b) What mutation score were you able to achieve in Lab1's test suite?

- Following result shows the mutation score achieved in Lab1's test suite:

```
---------------|---------|----------|-----------|------------|----------|--
--------|
File           | % score | # killed | # timeout | # survived | # no cov | #
errors |
---------------|---------|----------|-----------|------------|----------|--
--------|
All files      |   23.57 |       33 |         0 |        107 |        0 |
0 |
 calculator.js |    0.00 |        0 |         0 |        106 |        0 |
0 |
 lab1_main.js  |   97.06 |       33 |         0 |          1 |        0 |
0 |
---------------|---------|----------|-----------|------------|----------|--
--------|
```

(c) How do you kill those mutants?

- Following are the test cases for `month1`, `month2`, `day1`, `day2` and `year` conditional expressions:

```
test('main: month1', () => {
    assert.throws(() => Calculator.main(0, 1, 1, 1, 1), Error("invalid
month1"));
    assert.throws(() => Calculator.main(13, 1, 1, 1, 1), Error("invalid
month1"));
});

test('main: month2', () => {
    assert.throws(() => Calculator.main(1, 1, 0, 1, 1), Error("invalid
month2"));
    assert.throws(() => Calculator.main(12, 1, 13, 1, 1), Error("invalid
month2"));
});

test('main: day1', () => {
    assert.throws(() => Calculator.main(1, 0, 1, 1, 1), Error("invalid
day1"));
    assert.throws(() => Calculator.main(1, 32, 12, 1, 1), Error("invalid
day1"));
```

```
});

test('main: day2', () => {
    assert.throws(() => Calculator.main(1, 1, 1, 0, 1), Error("invalid
day2"));
    assert.throws(() => Calculator.main(1, 31, 1, 32, 1), Error("invalid
day2"));
});

test('main: year', () => {
    assert.throws(() => Calculator.main(1, 1, 1, 1, 0), Error("invalid
year"));
    assert.throws(() => Calculator.main(1, 1, 1, 31, 10001), Error("invalid
year"));
});
```

- For `month1 === month2 && day1 > day2` and `month1 > month2`:

```
test('main: month1 === month2 && day1 > day2', () => {
    assert.throws(() => Calculator.main(1, 2, 1, 1, 1), Error("day1 must be
less than day2 if month1 is equal to month2"));
    assert.throws(() => Calculator.main(1, 2, 1, 1, 10000), Error("day1
must be less than day2 if month1 is equal to month2"));
});

test('main: month1 === month2 && day1 === day2', () => {
    assert.strictEqual(Calculator.main(1, 1, 1, 1, 1), 0);
});

test('main: month1 > month2', () => {
    assert.throws(() => Calculator.main(2, 2, 1, 1, 1), Error("month1 must
be less than month2"));
});
```

- For `calculate`:

```
test('calculate', () => {
    assert.strictEqual(Calculator.main(2, 1, 4, 1, 400), 60);
    assert.strictEqual(Calculator.main(2, 1, 4, 1, 500), 59);
    assert.strictEqual(Calculator.main(1, 1, 12, 31, 1), 364);
    assert.strictEqual(Calculator.main(1, 1, 12, 31, 8), 365);
});
```

(d) What mutation score were you able to achieve?

```
---------------|---------|----------|-----------|------------|----------|--
--------|
```

```
File             | % score | # killed | # timeout | # survived | # no cov | #
errors |
---------------|---------|----------|-----------|------------|----------|--
--------|
All files        |   99.29 |      138 |         1 |          1 |        0 |
0 |
 calculator.js |  100.00 |      105 |         1 |          0 |        0 |
0 |
 lab1_main.js   |   97.06 |       33 |         0 |          1 |        0 |
0 |
---------------|---------|----------|-----------|------------|----------|--
--------|
```

(e) Can your test achieve 100% mutation score? Why not if you can't?

- I got 100.00% mutation score but 1 timout mutant: `for (let i = month1 + 1; i <= month2 - 1; i--)`
- Why can't I kill this mutant?
    - since `month1` will never be greater than `month2` in the test cases, so this mutant cannot be killed.