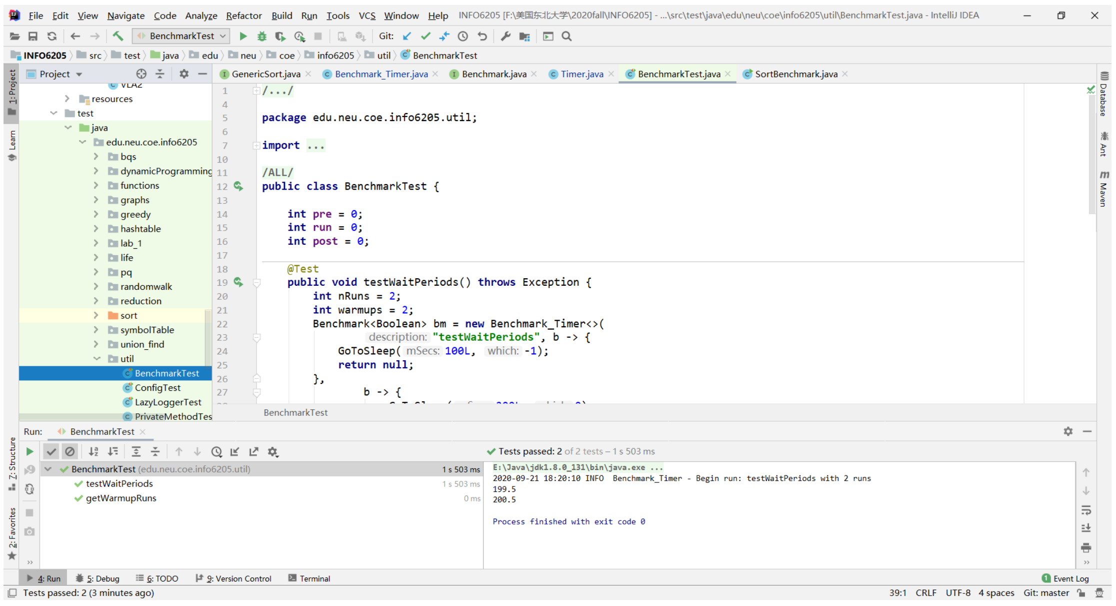
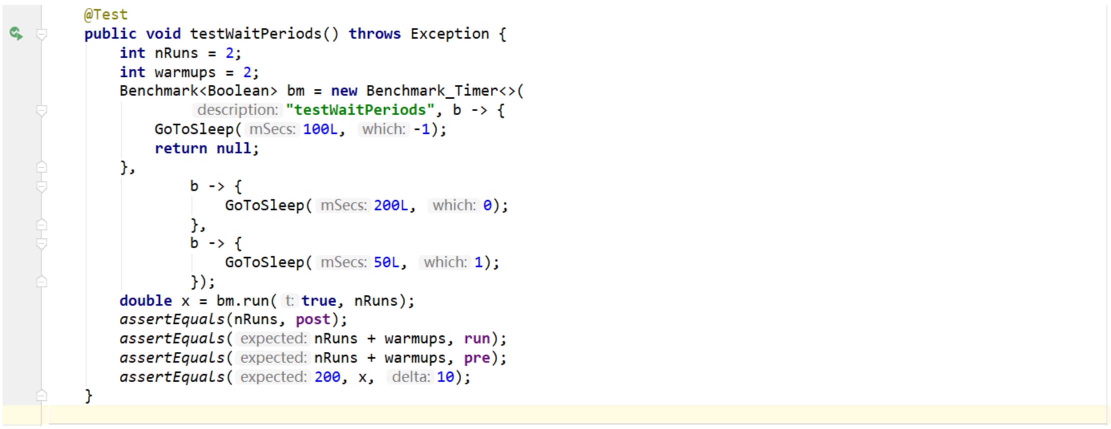


INFO6205 [F:\美国东北大学\2020fall\INFO6205] - ...\src\test\java\edu\neu\coe\info6205\util\BenchmarkTest.java - IntelliJ IDEA

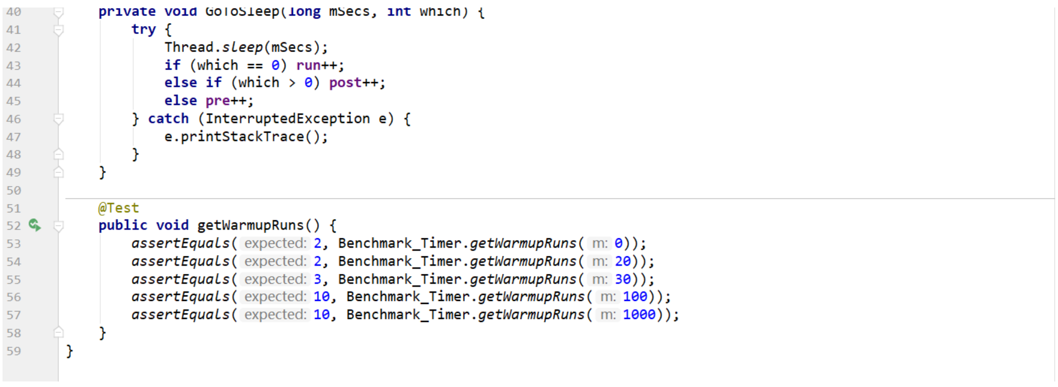
```java
/.../

package edu.neu.coe.info6205.util;

import ...

/ALL/
public class BenchmarkTest {

    int pre = 0;
    int run = 0;
    int post = 0;

    @Test
    public void testWaitPeriods() throws Exception {
        int nRuns = 2;
        int warmups = 2;
        Benchmark<Boolean> bm = new Benchmark_Timer<>(
                 description: "testWaitPeriods", b -> {
            GoToSleep( mSecs: 100L,  which: -1);
            return null;
        },
                b -> {
```

```
Tests passed: 2 of 2 tests – 1 s 503 ms

BenchmarkTest (edu.neu.coe.info6205.util)   1 s 503 ms
    testWaitPeriods                         1 s 503 ms
    getWarmupRuns                                 0 ms

E:\Java\jdk1.8.0_131\bin\java.exe ...
2020-09-21 18:20:10 INFO  Benchmark_Timer - Begin run: testWaitPeriods with 2 runs
199.5
200.5

Process finished with exit code 0
```

Tests passed: 2 (3 minutes ago)



```java
    @Test
    public void testWaitPeriods() throws Exception {
        int nRuns = 2;
        int warmups = 2;
        Benchmark<Boolean> bm = new Benchmark_Timer<>(
                 description: "testWaitPeriods", b -> {
            GoToSleep( mSecs: 100L,  which: -1);
            return null;
        },
                b -> {
                    GoToSleep( mSecs: 200L,  which: 0);
                },
                b -> {
                    GoToSleep( mSecs: 50L,  which: 1);
                });
        double x = bm.run( t: true, nRuns);
        assertEquals(nRuns, post);
        assertEquals( expected: nRuns + warmups, run);
        assertEquals( expected: nRuns + warmups, pre);
        assertEquals( expected: 200, x,  delta: 10);
    }
```



```java
    private void GoToSleep(long mSecs, int which) {
        try {
            Thread.sleep(mSecs);
            if (which == 0) run++;
            else if (which > 0) post++;
            else pre++;
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }

    @Test
    public void getWarmupRuns() {
        assertEquals( expected: 2, Benchmark_Timer.getWarmupRuns( m: 0));
        assertEquals( expected: 2, Benchmark_Timer.getWarmupRuns( m: 20));
        assertEquals( expected: 3, Benchmark_Timer.getWarmupRuns( m: 30));
        assertEquals( expected: 10, Benchmark_Timer.getWarmupRuns( m: 100));
        assertEquals( expected: 10, Benchmark_Timer.getWarmupRuns( m: 1000));
    }
}
```