

STIWK3014 REAL TIME PROGRAMMING

Tutorial / Exercise 8: TestAtomicInteger1p.java & Synchronization.java

Tasks 1:

- TestAtomicInteger1p.java failed to produce the correct result. Modify the TestAtomicInteger1p.java below to display "Count=10".
- Kindly show in your line of code for each modification as “//New modification “
//Screen shot the modification of the sample code and the output for the submission.
(Rename as: TestAtomicInteger1p.java Code Modification and Output)



```
package Week_07;

class TestAtomicInteger1p {

    public static void main(String[] args) throws InterruptedException {

        CountProblem pt = new CountProblem();
        Thread t1 = new Thread(pt, "t1");
        Thread t2 = new Thread(pt, "t2");
        t1.start();
        t2.start();
        t1.join();
        t2.join();
        System.out.println("Count=" + pt.getCount());
    }
}

class CountProblem implements Runnable {

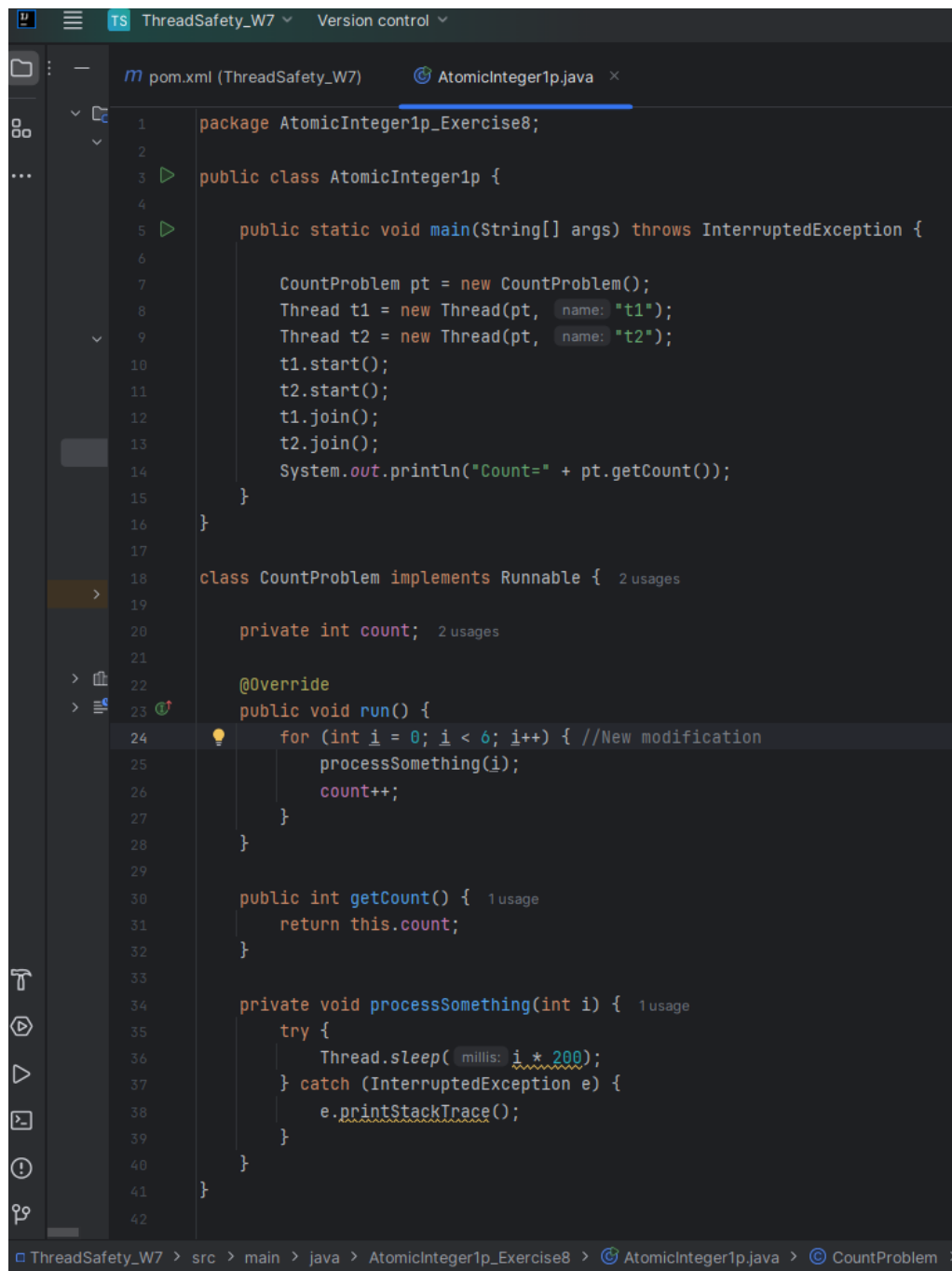
    private int count;

    @Override
    public void run() {
        for (int i = 1; i <= 5; i++) {
            processSomething(i);
            count++;
        }
    }

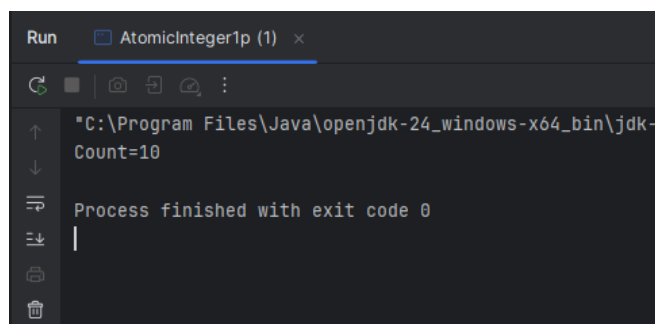
    public int getCount() {
        return this.count;
    }

    private void processSomething(int i) {
        try {
            Thread.sleep(i * 200);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
```

New Modification and Output



```
1 package AtomicInteger1p_Exercise8;
2
3 public class AtomicInteger1p {
4
5     public static void main(String[] args) throws InterruptedException {
6
7         CountProblem pt = new CountProblem();
8         Thread t1 = new Thread(pt, name: "t1");
9         Thread t2 = new Thread(pt, name: "t2");
10        t1.start();
11        t2.start();
12        t1.join();
13        t2.join();
14        System.out.println("Count=" + pt.getCount());
15    }
16 }
17
18 class CountProblem implements Runnable { 2 usages
19
20     private int count; 2 usages
21
22     @Override
23     public void run() {
24         for (int i = 0; i < 6; i++) { //New modification
25             processSomething(i);
26             count++;
27         }
28     }
29
30     public int getCount() { 1 usage
31         return this.count;
32     }
33
34     private void processSomething(int i) { 1 usage
35         try {
36             Thread.sleep( millis: i * 200);
37         } catch (InterruptedException e) {
38             e.printStackTrace();
39         }
40     }
41 }
42
```



```
Run AtomicInteger1p (1) x
C:\Program Files\Java\openjdk-24_windows-x64_bin\jdk-
Count=10
Process finished with exit code 0
```

Plagiarism

No mark will be given for plagiarism activities especially those copied from ChatGPT.

Submission:

Platform: 1. Online Learning - Sample Coding & Output in PdF form
2. GitHub – Upload the file and attach your GitHub link repositories.

Date: 11 May 2025 (Sunday, before 12.30 noon)