

COS40003 – Concurrent Programming

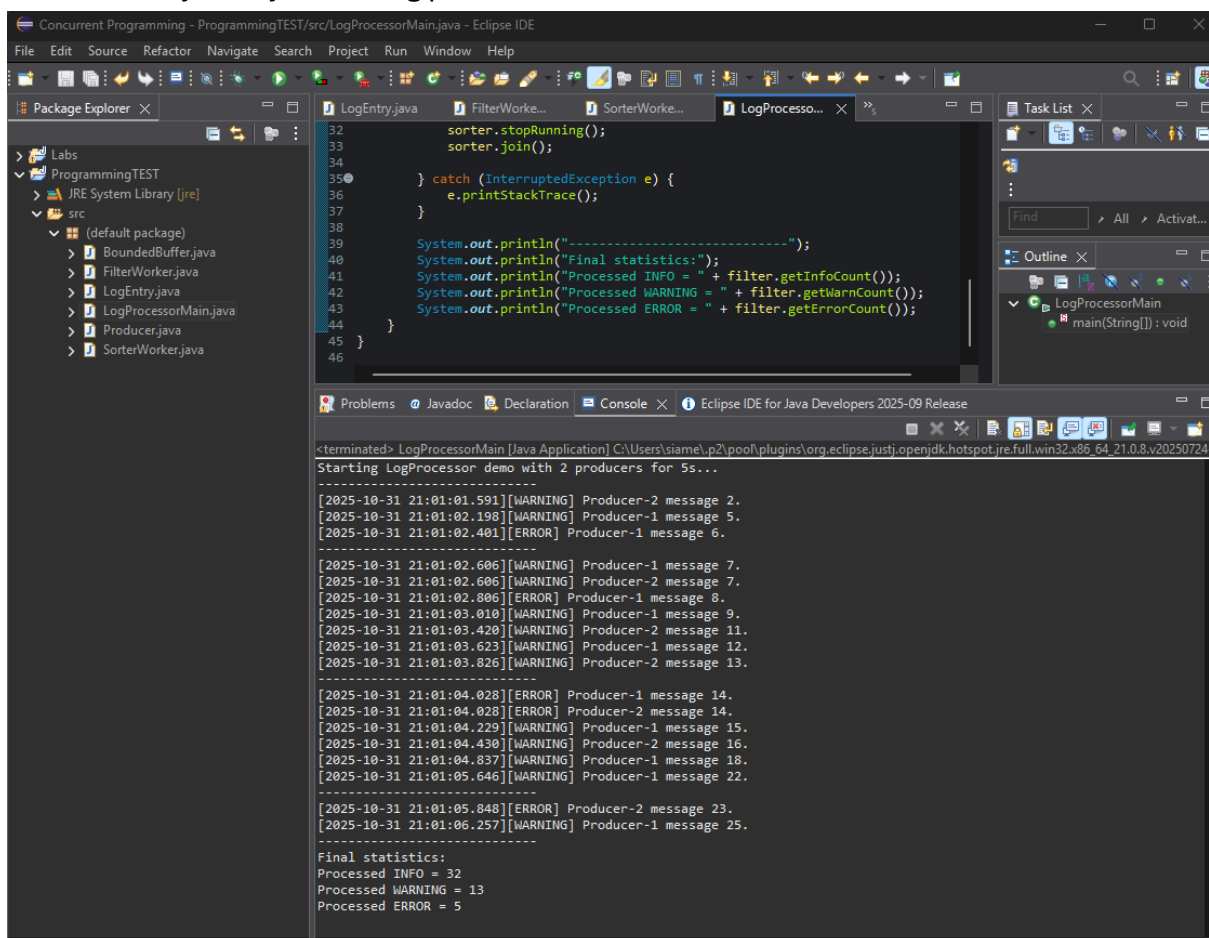
Programming Test, Semester 2, 2025

Name: Nur E Siam

Student ID: 103842784

Here is the output of the code correctly running displaying log entries generated by two producer threads, where only **WARNING** and **ERROR** messages are shown after being filtered and sorted chronologically.

The **INFO** messages are counted internally (totalling 32) but not displayed, as they are filtered out by the system's log processor.



The screenshot shows the Eclipse IDE interface. The Package Explorer on the left shows the project structure: Labs > ProgrammingTEST > JRE System Library [jre] > src > (default package) > BoundedBuffer.java, FilterWorker.java, LogEntry.java, LogProcessorMain.java, Producer.java, and SorterWorker.java. The Editor window shows the LogProcessorMain.java file with the following code:

```
32 sorter.stopRunning();
33 sorter.join();
34
35 } catch (InterruptedException e) {
36     e.printStackTrace();
37 }
38
39 System.out.println("-----");
40 System.out.println("Final statistics:");
41 System.out.println("Processed INFO = " + filter.getInfoCount());
42 System.out.println("Processed WARNING = " + filter.getWarnCount());
43 System.out.println("Processed ERROR = " + filter.getErrorCount());
44 }
45 }
46
```

The Console window at the bottom shows the output of the LogProcessorMain application. It starts with "Starting LogProcessor demo with 2 producers for 5s...". The output shows a series of log entries from two producers, filtered to show only WARNING and ERROR messages. The messages are sorted chronologically. The final statistics are:

```
Final statistics:
Processed INFO = 32
Processed WARNING = 13
Processed ERROR = 5
```