Ahsanullah University of Science and Technology

Department of Computer Science and Engineering

Program: Bachelor of Science in Computer Science and Engineering

Course No: CSE 3214

Course Title: Operating System Lab

Assignment No: 02

Date of Submission: 23.05.2024

Submitted to:

Mr. Md. Moinul Hoque **Associate Professor** **Department of CSE**	**Mr. Saha Reno** **Assistant Professor** **Department of CSE**

Submitted by:
Name: Afia Fahmida
Student ID: 20210104032
Section: A2

Reader-Writer Problem

Implementation in Java:

Main Class:

```
package readerwriter.assignment2;
public class Assignment2 {
    public static void main(String[] args) {
      SharedResource sharedResource = new SharedResource();
     System.out.println("Data in Buffer : " + sharedResource.getData());

        Thread[] readers = new Thread[3];
    for (int i = 0; i < readers.length; i++) {
        readers[i] = new Thread(new Reader(sharedResource), "Reader " + (i + 1));
        readers[i].start();
    }

    Thread[] writers = new Thread[2];
    for (int i = 0; i < writers.length; i++) {
        writers[i] = new Thread(new Writer(sharedResource), "Writer " + i);
        writers[i].start();
    }


    try {
        Thread.sleep(30000);
         } catch (InterruptedException e) {
        Thread.currentThread().interrupt();
    }

    for (Thread reader : readers) {
        reader.interrupt();
    }
    for (Thread writer : writers) {
        writer.interrupt();
    }

    System.out.println("All threads have finished execution.");
   }
}
```

SharedResorces Class:

```
package readerwriter.assignment2;
import java.util.ArrayList;
import java.util.List;
import java.util.concurrent.Semaphore;


public class SharedResource {
  private final Semaphore m = new Semaphore(1);
  private final Semaphore r = new Semaphore(1);
  private final Semaphore w = new Semaphore(1);
  private int readCount = 0;
  private List<Integer> data = new ArrayList<>(List.of(1, 5, 6, 7, 8, 9, 15));

public void read() {
   try {
      r.acquire();
      m.acquire();
      readCount++;

       if (readCount == 1) {
         w.acquire();
        }
      m.release();
      r.release();

      int index = (int) (Math.random() * data.size());
      System.out.println(Thread.currentThread().getName() + ": Reading " + data.get(index));

      m.acquire();
      readCount--;
       System.out.println(Thread.currentThread().getName() + ": leaves ");

     if (readCount == 0) {
         w.release();
      }


      m.release();
   } catch (InterruptedException e) {
      Thread.currentThread().interrupt();
   }
}
public void write(List<Integer> newData) {
    try {
        while (true) {
            m.acquire();
            if (readCount == 0) {
                m.release();
                break;
            }
            m.release();
            Thread.sleep(500);        }

        w.acquire();
          System.out.println(Thread.currentThread().getName() + " : writes in the system " +
newData);

        data.addAll(newData);

        System.out.println("\n ");
        System.out.println("Data in Buffer : " + data);

        w.release();
        System.out.println(Thread.currentThread().getName() + " : leaves the system");

         } catch (InterruptedException e) {


       Thread.currentThread().interrupt();
    }
}

public String getData() {
    return data.toString();
   }

}
```

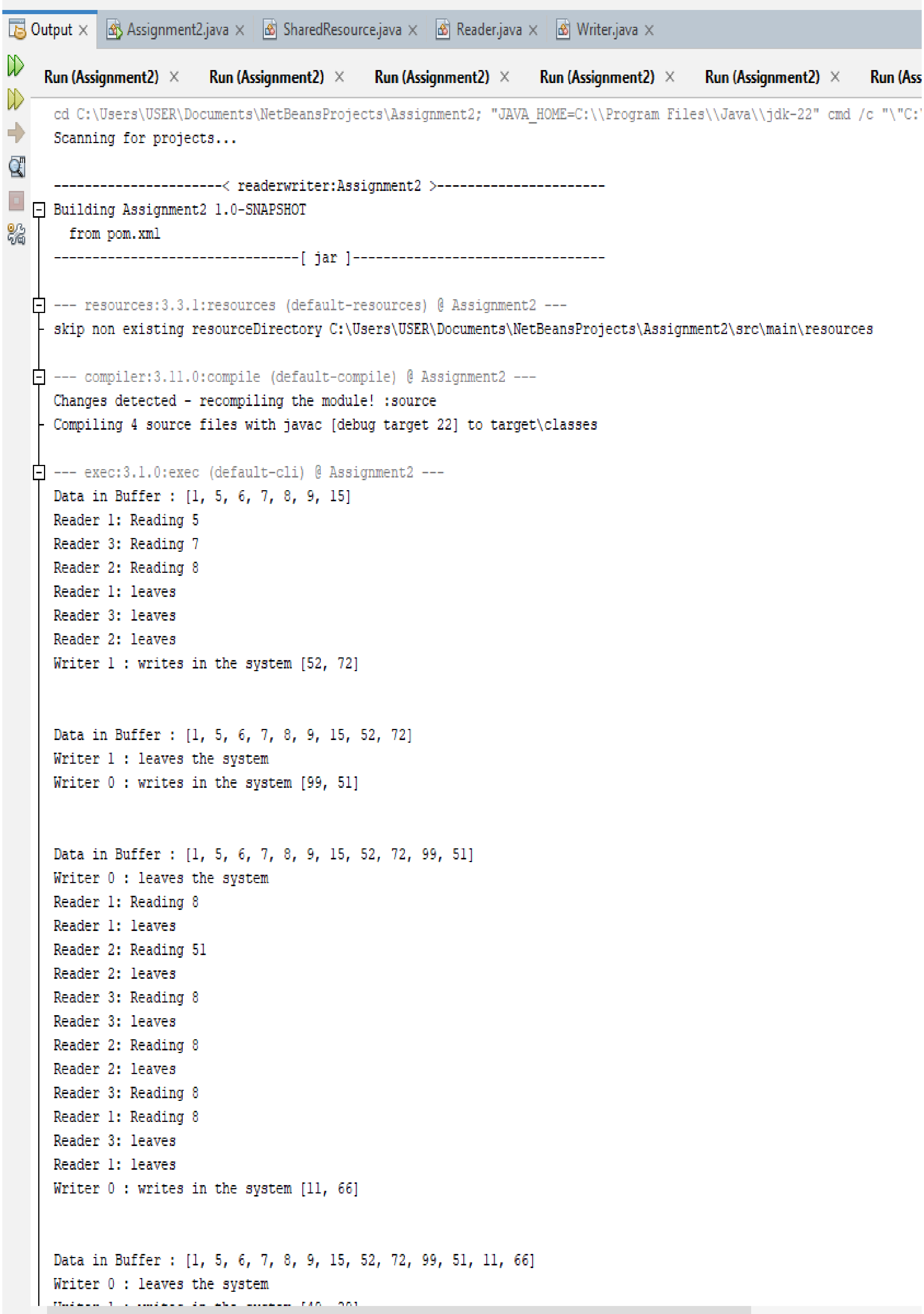
Reader Class:

```
package readerwriter.assignment2;
public class Reader implements Runnable {
        private final SharedResource sharedResource;
        public Reader(SharedResource sharedResource) {
                this.sharedResource = sharedResource; }
@Override
public void run() {
    while (true) {
      sharedResource.read();
      try { Thread.sleep(1000);
           } catch (InterruptedException e) {

     Thread.currentThread().interrupt(); break;
       }
    }
   }
}
```

Writer Calss:

```
package readerwriter.assignment2;
import java.util.ArrayList;
import java.util.List;
public class Writer implements Runnable {
    private final SharedResource sharedResource;
public Writer(SharedResource sharedResource) {
   this.sharedResource = sharedResource;
}

@Override
public void run() {
   while (true) {
      List<Integer> newData = new ArrayList<>();
      newData.add((int) (Math.random() * 100));
      newData.add((int) (Math.random() * 100));
      sharedResource.write(newData);
      try {
         Thread.sleep(2000);
      } catch (InterruptedException e) {
         Thread.currentThread().interrupt();
         break;
      }
   }
}
```

Output:



```
cd C:\Users\USER\Documents\NetBeansProjects\Assignment2; "JAVA_HOME=C:\\Program Files\\Java\\jdk-22" cmd /c "\"C:'
Scanning for projects...

----------------------< readerwriter:Assignment2 >----------------------
Building Assignment2 1.0-SNAPSHOT
  from pom.xml
--------------------------------[ jar ]---------------------------------

--- resources:3.3.1:resources (default-resources) @ Assignment2 ---
skip non existing resourceDirectory C:\Users\USER\Documents\NetBeansProjects\Assignment2\src\main\resources

--- compiler:3.11.0:compile (default-compile) @ Assignment2 ---
Changes detected - recompiling the module! :source
Compiling 4 source files with javac [debug target 22] to target\classes

--- exec:3.1.0:exec (default-cli) @ Assignment2 ---
Data in Buffer : [1, 5, 6, 7, 8, 9, 15]
Reader 1: Reading 5
Reader 3: Reading 7
Reader 2: Reading 8
Reader 1: leaves
Reader 3: leaves
Reader 2: leaves
Writer 1 : writes in the system [52, 72]


Data in Buffer : [1, 5, 6, 7, 8, 9, 15, 52, 72]
Writer 1 : leaves the system
Writer 0 : writes in the system [99, 51]


Data in Buffer : [1, 5, 6, 7, 8, 9, 15, 52, 72, 99, 51]
Writer 0 : leaves the system
Reader 1: Reading 8
Reader 1: leaves
Reader 2: Reading 51
Reader 2: leaves
Reader 3: Reading 8
Reader 3: leaves
Reader 2: Reading 8
Reader 2: leaves
Reader 3: Reading 8
Reader 1: Reading 8
Reader 3: leaves
Reader 1: leaves
Writer 0 : writes in the system [11, 66]


Data in Buffer : [1, 5, 6, 7, 8, 9, 15, 52, 72, 99, 51, 11, 66]
Writer 0 : leaves the system
Writer 1 : writes in the system [40, 20]
```