```java
/*
    Program Loader: loads in the program and stores it in memory
 */
package simple_computer_simulation;

import java.io.File;
import java.io.FileNotFoundException;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import java.util.logging.Level;
import java.util.logging.Logger;

/**
 *
 * @author bhitt
 */
public class ProgramLoader {
    //Properties
    private String displayName;
    private Integer firstWord;
    private Integer firstInstruction;
    private List<Integer> instructions;
    //Default Constructor
    ProgramLoader(){
        //instantiate list
        instructions = new ArrayList<>();
        //Read in from the file "SimpleComputer_Program1.txt
        File programFile = new File ("SimpleComputer_Program1.txt");
        //check to see the file exists
        if(!programFile.exists()){
```

```
32              System.out.println("The program file
'SimpleComputer_Program1.txt' does not exist.");
33          }else{
34              try {
35                  //read in from file
36                  Scanner fileIn = new Scanner
(programFile);
37                  //read in display name
38                  displayName = fileIn.nextLine();
39                  //display the program name
40                  System.out.println(displayName);
41                  //get the memory location where the
first program word is to be loaded
42                  firstWord = fileIn.nextInt();
43                      //System.out.println("First word
at memory location:"+firstWord);
44                  //get the memory location of the
first instuction to be executed
45                  firstInstruction = fileIn.nextInt();
46                      //System.out.println("First
instruction to be executed at memory location:"
+firstInstruction);
47                  //read in the rest of the
instructions
48                  while(fileIn.hasNextLine()){
49                      //get a line
50                      String line = fileIn.nextLine();
51                      //System.out.println(line);
52                      line = line.split("/")[0];
53                      //System.out.println(line);
54                      String[] numbers = line.split("
");
55                      for(int i=0;i<numbers.length;
i++){
```

```
56                              if(!numbers[i].isEmpty()){
57                                  instructions.add
(Integer.parseInt(numbers[i]));
58                              }
59                          }
60                      }
61
62                      //output list TESTING PURPOSES
63 //                     for(int i=0;i<instructions.size();
i++){
64 //                         System.out.println("["+i+"] :"
+instructions.get(i));
65 //                     }
66                  } catch (FileNotFoundException ex) {
67                      Logger.getLogger(ProgramLoader.
class.getName()).log(Level.SEVERE, null, ex);
68                  }
69              }
70      }
71      //Accessors
72      Integer getFW(){
73          return firstWord;
74      }
75      Integer getFI(){
76          return firstInstruction;
77      }
78      List<Integer> getInstruction(){
79          return instructions;
80      }
81 }
82
```