CS62: Spring 2025 | Lecture #3 (More Java, File I/O) worksheet | Jingyi Li

1. Consider the following code:

```
int i = 10;
int n = i++%5;
```

a. What are the values of i and n after the code is executed?

b. What are the final values of i and n if instead of using the postfix increment operator (i++), you use the prefix version (++i)? That is, the code was:

```
int i = 10;
int n = ++i%5;
```

2. What does this print?

```
int n1 = 10;
int n2 = 47;
int n3 = 4;
System.out.println((n1%n3>n2%n3) ? (n1+n2):(n1-n2));
```

3. Declare and initialize an array of strings with all the classes you are taking this semester.

Write for loop that loops through the class. If the class is called “CS62”, you need to bring “CS62: this is the best class ever, no need to see more” and break out of the loop. (Hint: use the .equals() method to compare Strings – `"hello".equals("hello")` is `true`)

Otherwise, if a class is called “CS101” you need to print “CS101: new CS achievement unlocked” and continue to the next iteration.

Otherwise, print the name of the class.

4. Fill in the missing lines.

```
import java.io.File;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Scanner;

public class FileIOExample {
    public static void main(String[] args) {
        Scanner inputScanner = new Scanner(System.in);
        System.out.println("Enter the input file name: ");
        String inputFile = inputScanner.nextLine();
        System.out.println("Enter the number of lines to read: ");
        String linesInput = _________;
        int numberOfLines = -1;
        try {
            numberOfLines = Integer.parseInt(________);
        } catch (NumberFormatException e) {
            System.err.println("_______________________");
        }
        finally{
            inputScanner._______; //close the scanner
        }
        Scanner fileScanner = null;
        PrintWriter writer = null;
        try {
            fileScanner = new ______________________________;
            writer = new ___________("output.txt");

            int linesRead = 0;
            //while there are still lines and we've read less than the input
            while (fileScanner.__________ && linesRead < _________) {
                String line = fileScanner.nextLine();
                System.out.println(line);
                writer.println(_____);
                _______++; //increment # of lines read
            }
        } catch (______________) {
            System.err.println("Error reading or writing files: " +
            e.getMessage());
        }
        finally{
            if(fileScanner!=null){
                fileScanner.close();
            }
            if(writer!=null){
                _________________;
            }
        }
    }
}
```