Problem 1: Algorithm Implementation and Basic Java (30 points)

a) (10 points) The following code is supposed to display the average of five command-line integer arguments, but it doesn't display the correct value. What is the problem with this code? How to fix it in order for it to display the average of the five command line integer arguments?

```
public static void main (String[] args) {
   int sum = Integer.parseInt(args[0]);
   sum += Integer.parseInt(args[1]);
   sum += Integer.parseInt(args[2]);
   sum += Integer.parseInt(args[3]);
   sum += Integer.parseInt(args[4]);

   double average = sum/5;
   System.out.println(average);
}
```

b) **(5 points)** The following code command-line inputs are a weekday (Monday through Sunday), the weekday temperature in Fahrenheit, and the conversion scale. It displays the converted temperature (Kelvin or Celsius) for the weekday.

Which of the following statements executes the program properly displaying the converted temperature, choose all that apply?

c) (15 points) Translate the following pseudocode into a Java class called ValidateTriangle.

```
READ angle1
READ angle2
READ angle3
COMPUTE sum AS angle1 + angle2 + angle3
IF sum equals 180 AND angle1 > 0 AND angle2 > 0 AND angle3 > 0 THEN
 IF angle1 equals angle2 AND angle1 equals angle3 THEN
   DISPLAY Equilateral triangle
 ELSE
    IF angle1 equals angle2 OR angle1 equals angle3 OR angle2 equals angle3 THEN
      DISPLAY Isosceles triangle
    ELSE
      DISPLAY Scalene triangle
    ENDIF
  ENDIF
ELSE
 DISPLAY Triangle is not valid
ENDIF
```

Problem 2: Loops (30 points)

a) (8 points) The following code is intended to continuously get user integer input and print a message based on the values input. Specifically, if the integer is 1, print "create a file." If the integer is 2, print "open a file." If the integer is 3, print "close a file." If the integer is 4, print "delete a file." The program will end if the user inputs any other integer values. The program does not work as intended. What code do you need to add and where do you need to add this code to fix the program?

```
1 public class Loop1
2 {
3
    public static main (String[] args)
4
5
      while (true)
6
7
        int op = StdIn.readInt();
8
         if (op == 1)
9
           StdOut.println("create a file.");
10
         else if (op == 2)
           StdOut.println("open a file.");
11
12
         else if (op == 3)
           StdOut.println("close a file.");
13
14
         else if (op == 4)
15
           StdOut.println("delete a file.");
16
       }
17
     }
18 }
```

- b) (12 points) Write a snippet among the integers from 1 to 99, print all integers that are divisible by three. The printed integers should be separated in different lines and each line except the last one has 5 integers. Use a for loop for this question.
- c) (10 points) There is a bug in the following code. The error line is line _____. Without modifying the error line, we can fix the bug in this way: _____.

```
1 public class Loop2
2 {
3
    public static main (String[] args)
4
5
      int n = Integer.parseInt(args[0]);
6
      int i = 1;
      while ( i < n ) {
7
      for ( int j = 1; j < n; j++ ) {</pre>
8
        if ( j % i == 0 )
9
10
           System.out.print(j + " ");
11
        else
           System.out.print(0 + " ");
12
13
14
     System.out.println();
15
     i++;
16
      }
17
     System.out.println("There are " + i + " rows and "+ j + "columns");
18
19 }
```

Problem 3: Arrays (40 points)

a) (12 points) For each of the code segments below, indicate if there is a compile-time error, a runtime error, or no error. If there is an error, BRIEFLY describe what the error is and where or why it occurs.

Code segment	Compile time	Error Description
	(C) or Runtime	
	error (R) or No	
	Error?	
<pre>int[] a = new int[];</pre>		
<pre>int[] a = {1, 2, 3, 4, 5}; for(int i = 0; i < a.length; i++) {</pre>		
a[i+1] = a[i];		
}		
<pre>int [][] arr = [1,2,3],[4,5,6];</pre>		
<pre>int[] a = new int[10];</pre>		
<pre>for(int i = -5; i < 5; i++) {</pre>		
a[i] = i;		
}		
<pre>int [] [] arr = {{1,2,3},{4,5,6}};</pre>		
<pre>int[] a = new int[arr.length];</pre>		
<pre>for (int i = 0; i < arr.length; i++) {</pre>		
<pre>for(int j = 0; j < arr[i].length; j++) {</pre>		
a[i] = arr[i][j];		
}		
}		
<pre>int[] a = {1, 2, 3, 4, 5};</pre>		
<pre>for(int i = a.length; i >= 0; i) {</pre>		
<pre>System.out.println(a[i]);</pre>		
}		

b) (18 points) Give the output for each of the following code segments.

int[] nums = {3,6,1,0,1,4,2};
for(int i = 0; i < nums.length - 1; i++){
 if (nums[i] > nums[i + 1]){
 | System.out.print(i + " " + nums[i] + " ");
 }
}

```
int[] nums = {3,6,1,0,1,4,2};
for(int i = 3; i < nums.length - 1; i++){
    nums[i] = nums[i + 1];
}
for(int e : nums){
    System.out.print (e + " ");
}</pre>
```

c) (10 points) Write a segment of code

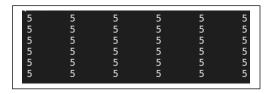
A 2D array of ints, nums, is declared and initialized to contain all 5s. nums has the same number of rows and columns. Write a segment of code that will modify the contents of the 2D array, nums, resulting with:

- The diagonal of nums being unchanged (contains 5s)
- The elements in nums that are BELOW the diagonal contain 0s
- The elements in nums ABOVE the diagonal contain 1s.

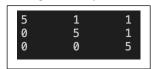
Examples:

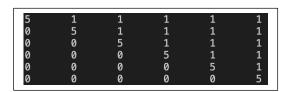
Original 2D array





Resulting 2D array





Problem 4: Input and Output (30 points)

a) (10 points) Write a Java program called SumAve that takes a variable number of doubles from the command line and displays their sum and their average. For example:

```
java SumAve 7 3 4 will display 14 4.6 java SumAve 11 8 5 20 will display 44 11
```

b) **(5 points)** The following code is supposed to sum two integer numbers but it does not work as intended. What is wrong with this code?

```
public class FindBug
{
   public static void main(String[] args)
   {
      // Add two numbers
      int a = Integer.parseDouble(args[0]);
      int b = Integer.parseDouble(args[1]);

      System.out.println("The sum is : " + (a+b));
   }
}
```

3. (15 points) What is the output?

```
Given the input File numin.dat:
     1 2 3 4 5 6 7 8 9 10
    11 12 13 14 15 16 17 18 19 20
    21 22 23 24 25 26 27 28 29 30
    31 32 33 34 55
 and the Java code InputArray. java:
   public class InputArray
      public static void main(String[] args)
       // read data into an array of integers
       int[] intArray = StdIn.readAllInts();
       for (int i = 0; i < intArray.length; i++)</pre>
       {
           if (intArray[i] % 11 == 0)
             System.out.print("Check ");
       }
       StdOut.println(intArray[args.length]);
      We compile to bytecode (javac InputArray.java) and run using redirection:
      <Linux/Mac> : java InputArray < numin.dat</pre>
      <Windows> : cat numin.dat | java InputArray
     What is the output of the program?
```