Problem 1: Data Types (14 points)

1. Evaluate the following expression and select the type and value of the result

   (int) 102.78

   a. int, 102
   b. double, 102
   c. double, 102.0
   d. long, 102
   e. Compile Error

2. Evaluate the following expression and select the type and value of the result

   (11 < 5) || (2 == 2) || (3 == '3')

   a. boolean, true
   b. boolean, false
   c. int, true
   d. Compile Error
   e. Runtime Error

3. Evaluate the following expression and select the type and value of the result

   (4 == 2) || ((5 != 5) && (1 < 8))

   a. boolean, true
   b. boolean, false
   c. int, true
   d. Compile error

4. Evaluate the following expression and select the type and value of the result
   (35 % 14)

   a. double, 7.0
   b. double 2.5
   c. int, 2
   d. int, 7
   e. Compile error

5. Evaluate the following expression and select the type and value of the result

   (3 || (2 != 33))

a. boolean, true
b. boolean, false
c. int, true
d. int, false
e. Compile error

6. What is the output of `System.out.println("4" + 6);`
   a. 10
   b. 46
   c. four6
   d. None of the above

7. What is the output of the following snippet of code?

```
int a = 3;
System.out.println( a + a );
```

   a. true
   b. 6
   c. Compile error
   d. aa

8. The following statement is incorrect. Select the appropriate statement.

```
int b = args[0];
```

   a. int b = (int) args[0];
   b. int b = parseInt(args[0]);
   c. int b = Integer.parseInt(args[0]);
   d. None of the above

9. Give the type and value for the expression
`Integer.parseInt("3");`

   a. int, true
   b. String, 3
   c. int, 3
   d. Compile error
   e. int, 3.0

10. Evaluate the following expression and select the type and value of the result
(9 / 2) * 3.0

a. double, 13.5
b. double, 13.0
c. int, 13
d. double, 12.0
e. int, 12

11. Evaluate the following expression and select the type and value of the result
(9 / 2.0) * 2

a. double, 9.0
b. double, 8.0
c. int, 9
d. double, 12.0
e. int, 8

12. Evaluate the following expression and select the type and value of the result
1.11 + "CS"

a. double, 1.11
b. String, "1.11CS"
c. Compile error
d. Runtime error
e. None of the above

13. Evaluate the following expression and select the type and value of the result
(120 / 0) * 3

a. double, 0.0
b. int, 0
c. Compile error
d. Runtime error
e. None of the above

14. Evaluate the following snippet of code and select value of the result

```
int b = 4;
b += 6;
System.out.println(b);
```

a. 4
b. 46
c. Runtime error
d. 6
e. 10

Problem 2: Truth Table (16 points)

a) **(8 points)** Show the truth table for the Boolean expression: `!(x && !y && z) || y`

			Output
x	**y**	**z**	**!(x && !y && z) || y**
false	false	false	
false	false	true	
false	true	false	
false	true	true	
true	false	false	
true	false	true	
true	true	false	
true	true	true	

b) **(8 points)** Show the truth table for the Boolean expression: `(x && !y) || (y && !z)`

			Output
x	**y**	**z**	**(x && !y) || (y && !z)**
false	false	false	
false	false	true	
false	true	false	
false	true	true	
true	false	false	
true	false	true	
true	true	false	
true	true	true	

Problem 3: Pseudocode (35 points)

The pseudocode on the right below is intended to display the smallest of the three numbers a, b, and c. The table on the left below gives the **intended** results for the specified inputs of a, b, and c.

a	b	c	Intended Displayed Result	Actual Displayed Result
1	2	3	1 DONE	
2	1	3	1 DONE	
2	1	2	1 DONE	
2	2	2	2 DONE	
1	2	1	1 DONE	
1	2	2	1 DONE	

```
READ a
READ b
READ c
IF a < b AND a < c THEN
   DISPLAY a
ENDIF
IF b < a AND b < c THEN
   DISPLAY b
ENDIF
IF c < a AND c < b THEN
   DISPLAY c
ENDIF
DISPLAY DONE
```

a. **(12 points)** The pseudocode above does not work as intended in all cases. Complete the last column of the table above to indicate what is actually displayed by the pseudocode above.

b. Three students were tasked with developing a pseudocode solution that would solve the problem correctly in ALL cases. Their solutions appear below.

Student #1

```
READ a
READ b
READ c
IF b < a THEN
  SET num TO a
  SET a TO b
  SET b TO num
ENDIF
IF c < a THEN
  SET num TO a
  SET a TO c
  SET c TO num
ENDIF
IF c < b THEN
  SET num TO b
  SET b TO c
  SET c TO num
ENDIF
DISPLAY a
```

Student #2

```
READ a
READ b
READ c
IF (a <= b) THEN
       IF (a <= c) THEN
          DISPLAY a
       ELSE
          DISPLAY c
       ENDIF
ELSE
       IF (b <= c) THEN
          DISPLAY c
       ELSE
          DISPLAY b
       ENDIF
ENDIF
```

Student #3

```
READ a
READ b
READ c
SET num TO a
IF b < num THEN
   SET num to b
ELSE
   IF c < num THEN
       SET num TO c
   ENDIF
ENDIF
DISPLAY num
```

(18 points) For each solution above, indicate if the solutions will display the smallest of a, b, and c IN ALL CASES. If the solution does not work IN ALL CASES, give input values for a, b, and c that would demonstrate this (displaying an incorrect smallest value).

Solutions	Does the solution work **IN ALL CASES? (YES or NO)**	If not, give inputs for a, b, and c that demonstrate the solution will not work in all cases. Identify the inputs as a, b, and c. If the solution works in all cases, put "WORKS" in this column	Total number of operations in the student solution if the given input is a = 1, b = 3, c = 2
Student #1			
Student #2			
Student #3			

(5 points) Write a pseudocode solution to determine and display the smallest of FOUR numbers. Your code should work FOR ALL CASES.

Problem 4: Counting Operations (35 points)

a) The following pseudocode algorithm displays "Pay is greater than $200" if an employee makes more than $200 and displays "Pay is under $200" otherwise.

Input: hoursWorked and ratePerHour
Output: totalPay
Error: negative inputs

Pseudocode:

```
READ hoursWorked
READ ratePerHour
IF hoursWorked < 0 OR ratePerHour < 0 THEN
    DISPLAY ERROR
ELSE
    COMPUTE totalPay AS hoursWorked * ratePerHour
    IF totalPay > 200 THEN
        DISPLAY "Pay is greater than $200"
    ELSE
        DISPLAY "Pay is $200 or under"
    ENDIF
ENDIF
```

(5 points) What is the minimum number of operations that could be executed by this code? 5
(5 points) What is the maximum number of operations that could be executed by this code? 7

b) The following pseudocode algorithm displays the values 5 4 3 2 1

Input: none
Output: 5 4 3 2 1
Pre-condition: none

Pseudocode:

```
SET count TO 5
SET end TO 1
WHILE count >= end
    DISPLAY count
    SUBTRACT 1 FROM count
ENDWHILE
```

(10 points) What is the total number of operations that are executed by this code?

c) The following pseudocode algorithm reads two integer values startInt and n, then it displays the sum of starInt and the next n consecutive integers.

For example:

-if `startInt` is 7 and n is 4 the program displays 45 as 7 + 8 + 9 + 10 + 11 = 45 (the sum of 7 and the 4 integers after it).

-If `startInt` is 9 and n is 7 the program displays 100 as 9 + 10 + 11 + 12 + 13 + 14 + 15 + 16 = 100 (the sum of 9 and the 7 integers after it).

Input: Two integers
Output: the sum of the integer and the n numbers after it.
Pre-condition: A positive integer as input

Pseudocode:

```
READ startInt
READ n
SET num TO startInt
SET sum TO 0
SET end TO startInt + n

WHILE num <= end
    COMPUTE sum AS sum + num
    ADD 1 TO num
ENDWHILE

DISPLAY sum
```

(15 points) What is the total number of operations executed by this code in terms of n?

Problem 5: Java Programming (30 points)

a) **(20 points)** The following algorithm calculates the total miles `total` that Tom runs each week. The input `day` is an integer used to count the total number of days that Tom runs each week. The input `miles` is a floating-point number used to record the number of miles that Tom runs a day, namely, miles can contain decimal. Write the corresponding Java code.

```
READ day
READ miles
COMPUTE total AS day * miles
DISPLAY total
```

b) **(10 points)** The code fragment below is intended to behave according to the following rules but it doesn't. Correct the program so that it has no errors and prints the intended output. Suppose that `a`, `b` and `c` are distinct numbers.

Input	Output
`a > b and b > c`	`descending true` `ascending false`
`a < b and b < c`	`descending false` `ascending true`

```
1.   public static void main (String[] args) {
2.
3.     int a = Integer.parseInt(args[0]);
4.     int b = Integer.parseInt(args[1]);
5.     int c = Integer.parseInt(args[2]);
6.     boolean p = ( a > b ) || (b > c);
7.     boolean q = ( b < a ) || (b < c);
8.
9.     System.out.println("descending " + p);
10.    System.out.println("ascending " + q);
11.  }
```