Introduction to Computer Science - CS111
Fall 2021 Exam 1

Problem 1: Data Types (14 points)

1. Which of the following choices is the correct syntax for declaring a real number variable named 'points’ and initializing its value to 10.0?

    a. points = 10;
    b. double points = 10.0;
    c. points = double 10.0;
    d. int points : 10.0;
    e. 10.0 = points;

2. Which, if any, of the following is NOT a primitive type in Java?

    a. int
    b. double
    c. String
    d. char
    e. All of the above are Java primitive types.

3. Assume that A and B are Boolean variables and have been properly initialized. Consider the Boolean expression: ! (!A && !B) . Which of the following always evaluates to the same value as the expression above?

    a. b (!A && !B)
    b. (!A || !B)
    c. (A || B)
    d. ! (A || B)
    e. (!A || B)

4. Assume that **x** and **y** are Boolean variables and have been properly initialized. Consider the Boolean expression: **(x || y) && x** . Which of the following always evaluates to the same value as the expression above?

    a. x
    b. y
    c. x && y
    d. x || y

e. x != y

Problem 1: Data Types (short answer 20 points)

A. (12 points) Evaluate the following expressions. Give the **type** and **value** of each (2 points each = 12 points).

Expression	**Type**	**Value**
(8 / 3) + 2.0	double	4.0
492 / 10 % 10 / 2	int	4
53 / 5 / (0.6 + 1.4) / 2 + 13 / 2	double	8.5
“Computer 32” + 2 * 3	String	Computer 326
“(int)2.0” + 2 * 5	String	(int)2.010
4 + (-3 + 2) > -3 – (-3 + 4) * 2	boolean	true

B. (2 points) Imagine you are writing a Java program that stores a student's year (Freshman, Sophomore, Junior, or Senior), the number of courses the student is taking, and his or her GPA on a 5.0 scale. Declare variables with the appropriate names and types to hold this information. Write complete variable declaration statements with the type, the variable name, and a semicolon.

String year;
int numCourses;
double gpa;

C. (8 Points) Truth Table: Complete the truth table for the Boolean Expression:

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

true	false	false	true
false	true	true	false
false	true	false	false
false	false	true	false
false	false	false	false

Problem 2: (35 points) Given a problem write an algorithm using Pseudocode (no loops)

A. (10 points) Write a program in pseudocode that reads a 3-digit number and determines if the number is a palindrome or not. A number is a palindrome if it reads the same from left to right and right to left. For example, 121, 212 are palindromes, but 122 is not. You may use the operators / and % to complete this assignment. You are not allowed to use loops. The program must display the messages “Palindrome” or “not Palindrome”.

```
READ number
COMPUTE firstDigit as number / 100
COMPUTE thirdDigit as number % 10
IF (firstDigit == thirdDigit)
     DISPLAY “Palindrome”
ELSE
     DISPLAY “not Palindrome”
ENDIF
```

B. (25 points) *Tax Rate*. The amount of tax paid by a family will be determined based on several factors, including gross income earned by family (A), dividend income from investments (B), savings interest earned(C), home mortgage interest(D), number of dependents supported (E), and amount of charitable donations(F). In a hypothetical situation, we assume the following.

i. gross income, dividend, and savings interest are taxable

ii. the family is allowed to deduct $6000 per dependent from total income

iii. home mortgage interest is tax-deductible up to $10,000 (that is, any interest paid over $10,000 cannot be deducted)

iv. charitable donations up to 10% of gross income are tax-deductible.

v. dividend income from investments (B) is taxable at a fixed rate of 8%

vi. adjusted income is computed by subtracting deductions (D, E) from gross income + savings interest (A+C)

NOTES: The total taxes on adjusted income (adjusted income * tax Rate) are computed according to the following rules.

The first $50,000 of the adjusted income is computed at a tax rate of 8%
The next $100,000 of the adjusted income is computed at a tax rate of 20%
Excess adjusted income over $150,000 is computed at a tax rate of 33%

b. **Example 1.** if a family has an adjusted income of $175,000, then their tax is calculated as = 50,000*0.08 + 100,000*0.20 + (175,000-150,000)*0.33 + dividend tax

c. **Example 2.** a family with adjusted income of $82,000 will pay = 50000*0.08 +

(82000-50000)*0.20 in taxes + dividend tax.

d. **Example 3.** a family with adjusted income of $42,000 will pay = 42,000*0.08 in taxes + dividend tax. **

The taxes for dividends must be calculated separately and added to the tax total. Write a program in pseudocode to compute the total taxes for a family.

```
READ grossIncome
READ dividendIncome
READ savingsInterest
READ mortageInterest
READ numDependents
READ charitableDonations
SET totalTaxes to 0

COMPUTE totalIncome AS grossIncome + savingsInterest
IF mortageInterest > 10,000
SET mortgageInterest TO 10,000
ENDIF
IF charitableDonations > .10*grossIncome
     SET charitableDonations TO 0.10*grossIncome
ENDIF
COMPUTE deductibleAmount AS numDependents*6000 +
charitableDonations + mortageInterest
COMPUTE taxableIncome AS totalIncome - deductibleAmount
IF taxableIncome > 150000
     COMPUTE totalTaxes as totalTaxes +(taxableIncome -
150,000)*0.33
     SET taxableIncome AS taxableIncome – 150,000
ENDIF
IF taxableIncome > 50000
     COMPUTE totalTaxes as totalTaxes + (taxableIncome -
50,000)*0.20
     SET taxableIncome AS taxableIncome – 50,000
ENDIF
COMPUTE totalTaxes as totalTaxes + (taxableIncome)*0.08
COMPUTE totalTaxes AS totalTaxes + dividendIncome*0.08
DISPLAY totalTaxes
```

Problem 4: (35 points) Given a pseudocode algorithm, count the number of operations.

A. (10 points) The following pseudocode algorithm is used daily by a delivery company to check their cardboard boxes stock. The program finds the current stock of small, medium, and large boxes. Then, if the stock of each box size is below the given limit a new order is suggested.

```
READ smallBoxStock
READ mediumBoxStock
READ largeBoxStock
READ limit
SET numberOfSmallBoxes  TO 0
SET numberOfMediumBoxes TO 0
SET numberOfLargeBoxes  TO 0
SET totalOrder TO 0

IF smallBoxStock < limit THEN
     SET numberOfSmallBoxes TO 50000
     COMPUTE totalOrder TO totalOrder + numberOfSmallBoxes
ENDIF

IF mediumBoxStock < limit THEN
     SET numberOfMediumBoxes TO 50000
     COMPUTE totalOrder TO totalOrder +
numberOfMediumBoxesENDIF

IF largeBoxStock < limit THEN
     SET numberOfLargeBoxes TO 50000
     COMPUTE totalOrder TO totalOrder + numberOfLargeBoxes
ENDIF

DISPLAY numberOfSmallBoxes
DISPLAY numberOfMediumBoxes
DISPLAY numberOfLargeBoxes
DISPLAY totalOrder
```

a. (5 points) What is the minimum number of operations the code executes? 15
b. (5 points) What is the maximum number of operations the code executes? 21

B. (10 points) The following pseudocode counts the number of digits in a number.

Pseudocode:

```
READ number
SET counter TO 0
WHILE number > 0
ADD 1 TO counter
COMPUTE number AS number / 10
ENDWHILE
DISPLAY counter
```

Let x be the number of digits in a number. What is the number of operations the code executes in terms of *n*?

$2 + 3x + 1 + 1 = 3n + 4$

2 (READ, SET)

3x (3 operations in the loop, when number > 0 is true)

1 (when number>0 is false)

1 (DISPLAY)

C. (15 points) A lottery game works as follows: a customer selects a number *n* and is sold a ticket that has *n* numbers on it. The customer wins If all of the numbers on the ticket are the same. For example, if n = 3, a winning ticket would be 7 7 7.

The following pseudocode simulates this game, it reads a number *n* and then the numbers on the ticket, one at a time. Assume that *n* is at least 2.

For example: If a customer selects 3 the numbers input could be 3 1 2 4, or 2 1 (winner!), 4 1 2 4 3, 2 3 3 (Winner!), 3 7 7 7 (Winner!), etc…

a. (5 points) What is the minimum number of operations the code executes (enter an integer number only)? 7

b. (10 points) What is the maximum number of operations the code executes?

$3 + 6n + 1 + 1$

3 (READ, READ, SET) +

6n (6 operations in the loop, when counter <=n is true)

1 (when counter <=n is false)

1 DISPLAY

Problem 5: Java Programming (30 points)

a) (15 points) The pseudocode below takes an integer as input and prints the factorial of that number. Write the equivalent program, including taking the integer from the command line, in Java. Call the program MyFact.java

```
READ n
SET count TO n
SET fact TO 1
WHILE count > 0
COMPUTE fact AS fact * count
SUBRACT 1 FROM count
ENDWHILE
DISPLAY fact
```

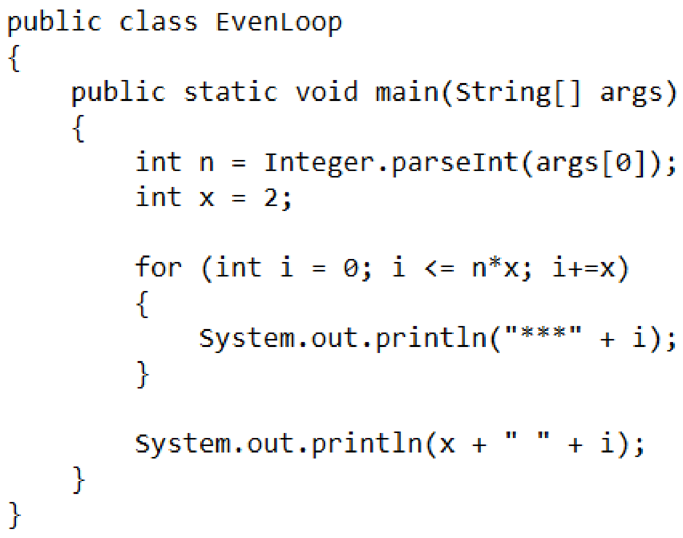
```
public class MyFact
{
    public static void main(String[] args)
    {
        int n = Integer.parseInt(args[0]);
        int fact = 1;
        int count = n;

        while ( count > 0)
        {
            fact = fact * count;
            count -= 1;
        }
        System.out.println(fact);
    }
}
```

b) (15 points) The following program does not behave as expected, it has a bug. Find the problem and explain why it is a bug. Then explain how to update the program so that it behaves as expected.

For example, suppose n = 4, the output of EvenLoop is supposed to be:

```
***0
***2
***4
***6
***8
2 8
```

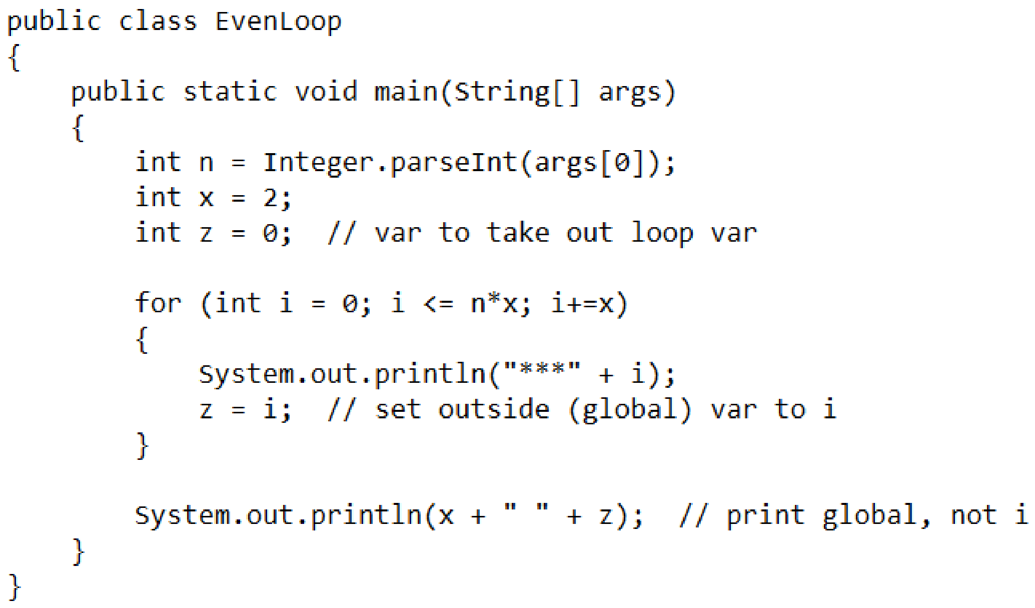


```
public class EvenLoop
{
    public static void main(String[] args)
    {
        int n = Integer.parseInt(args[0]);
        int x = 2;

        for (int i = 0; i <= n*x; i+=x)
        {
            System.out.println("***" + i);
        }

        System.out.println(x + " " + i);
    }
}
```

The problem is scope, the program above tries to print the loop variable outside of the loop.
10 pts if the student references scope (bad var, can't see, not inside loop, etc) but does not fix
12 pts if the student actually write the word "scope."
14 pts if the student fixes by using n*x instead of i in println
15 pts if they set a global var and use it outside loop.



```
public class EvenLoop
{
    public static void main(String[] args)
    {
        int n = Integer.parseInt(args[0]);
        int x = 2;
        int z = 0;  // var to take out loop var

        for (int i = 0; i <= n*x; i+=x)
        {
            System.out.println("***" + i);
            z = i;  // set outside (global) var to i
        }

        System.out.println(x + " " + z);  // print global, not i
    }
}
```