PRACTICE FOR QUIZ 1
CMPS 101 - Winter 02
Thomas Raffill

Name:__________________________
Student ID:____________________

This exam is closed book, closed notes, no electronic devices. Show all work. Partial credit given for partial solutions. Presentation counts! Be legible and coherent for full credit.

Question 1:___________ (out of 20)
Question 2:___________ (out of 20)
Question 3:___________ (out of 20)
Question 4:___________ (out of 20)
Question 5:___________ (out of 20)

Total:___________ (out of 100)
1. (20 points)
In the following statement of Java code, list the comment, keyword, identifier, literals, operators and punctuation.

```java
int product = 2*3*4*5; // Compute 5 factorial.
```

(a) Comment: // Compute 5 factorial.
(b) Keyword: int
(c) Identifier: product
(d) Literals: 2, 3, 4, 5
(e) Operators: =, *
(f) Punctuation: ;

2. (20 points)
Consider the following code fragment:

```java
int a = 5;
double b = a++ / 2 + a * a + 1.0;
```

(a) What is the value of a after this piece of code executes? 6
(b) What is the value of b after this piece of code executes? 39.0

Comment: Here is how the expression is evaluated: First, we apply the precedence rules to parenthesize the expression.

```
b = (((a++)/2) + (a*a)) + 1.0);
```

This parenthesization follows the operator precedences of ++ over /, the precedence of / and * over +, the precedence of all of them over =, and the left-to-right associativity of the + operator. Now we evaluate the expression from left to right. We want to assign to b the value of the expression on the right. First, we apply the suffix increment operator to a. This assigns the new value of 6 to a, but returns the value 5 to the expression. So we evaluate 5/2 with integer division. The result of this is 2, and a now has the value 6. Next, we evaluate a*a. This gives 6*6 or 36. Now we evaluate 2 + 36, and this gives 38. Finally we add 38 + 1.0, and since 1.0 is a double literal, we widen to a double before we assign to the double b, which in this case makes no difference and b is assigned the value 39.0.
3. (20 points)
What, if anything, does the following code fragment print out?

```java
int x = -4;
if (x > 0)
    if (x % 2 == 0)
        System.out.println("x is even.");
else
    System.out.println("x is negative.");
```

It prints out nothing.

**Comment:** This is an illustration of the “dangling else problem.” There are no block delimiters { or } around the statements after the if and else statements. Therefore, the else clause is associated with the second if and not the first if. When this code executes, first it checks whether x > 0. This is false, so it skips past the second if statement and its else clause, and as a result it prints out nothing. At first glance, it would seem that this code should print out “x is negative.” This is a common programming error.
4. (20 points)
Write a code fragment which uses a loop to print out the product of the first k integers.

```java
int product = 1;
for(int i=1; i<=k; i++) {
    product *= i;
}
System.out.println(product);
```

5. (20 points) Draw a flowchart for the following algorithm:

input an integer X
in a loop, do the following:
    print out X
    if X equals 1, exit the loop
    otherwise, if X is even set the value of X to X/2
    otherwise, set the value of X to (3*X) + 1
end of loop