

Name _____

Lab _____

Recall the Academic Integrity statement that you signed. Write all answers clearly on these pages, ensuring your final answers are easily recognizable. The number of points for each problem is clearly marked, for a total of 25 points. I will post my solutions on the web on Monday, off the **Solutions** link, after class.

1. (5 pts) In the following Java code, identify and classify every token: Draw a vertical line between each token and label every **identifier** with an **I**, every **keyword** with a **K**, every **separator** with an **S**, every **operator** with an **O**, every **literal** with an **L**, and every **comment** with a **C**. Write each letter below the box of the token that it classifies. For each literal, also indicate its type (**int**, **double**, **boolean**, **char**, or **String**). Look at the tokens exactly as the Java compiler would. These tokens are very close together: please draw the boxes as neatly as you can.

```
int x = Prompt.forInt("Enter",0,9); if (x<=5) {x++; System.out.print(""+x);} //close code
```

2. (2 pt) For each “local variable declaration” (a) if its syntax is **illegal** explain why; (b) if its syntax is **legal**, write the name of each variable initialized/changed and its resulting value. Recall that any simple or complicated expression (not just literals) can appear to the right of the = in the declarator’s optional initialization. In part (3) the two declarations come one right after the other; write the required information for each separately.

(1) `int x,y=3;`

(2) `int x=3.2;`

(3) `int y=0;`
`double x=y=3;`

3. (4 pts) Write a prototype that describes the operand and result types for each of the following methods. Choose good names. Do not attempt to explain **how** the method computes its result; just write its prototype.

a) A method that determines the number of fractions between two integers (inclusive) whose numerator and denominator are no larger than a than a third integer. For example, the number of fractions between 2 and 4 whose numerator and denominator are no larger than 5 is 5: 2/1, 3/1, 4/1, 4/2, 5/2).

b) A method that determines whether there are any primes between two integers inclusive. For example, there are primes between 15 and 20 (17 and 19 are both prime).

c) A method that selects those characters from one string that are specified in another (discarding the others). For example, when selecting from "Able I was, ere I saw Elba" using "aeiou" the result is "AeIaeEaEa" (case is unimportant for the selection process)

d) A method that determines whether or not two lines intersect. Each line is represented by two **double** values —its slope and its Y intercept.

4. (4 pts) Write the literal value resulting from applying each of the following operators to its operands.

a) `148 / 10`

b) `148 % 10`

c) `7 / 10`

d) `7. / 10`

e) `(double) (7/10)`

f) `(double) 7/10`

g) `'a' > 'z'`

h) `true != false`

i) `'4' + '1'`

j) `5+1+"1"+1`

5. (5 pts) Draw an oval diagram, with types (on top) and values (below) for the expression below. Assume `int x=3,y=2`; in your diagram, draw ovals around **every Java expression**. Please be neat.

`- 1 + 3 * x > 5 && Math.min(3 , y - 2) >= x`

6. (5 pts) Translate the following mathematical formulas into their equivalent Java expressions. Every italic letter is a variable. The symbol $|X|$ means the absolute value of X . Assume the following prototypes.

`double Math.pow (double,double)`

`double Math.abs (double)`

`double Math.sqrt (double)`

All variables are declared to be double. For these expressions, write no redundant parentheses!

(parentheses are redundant if they do not change the order in which operators are evaluated)

a) $\frac{5}{9}(f - 32)$ b) $\frac{|a-b|}{2(a+b)}$ c) $\frac{uir^2}{2(r^2 + x^2)^{\frac{3}{2}}}$ d) $\frac{m+c^2}{\sqrt{1+\frac{v^2}{c^2}}}$ e) Do **x**, **y**, and **z** store increasing values whose sum is 10 (**x**, **y**, and **z** storing 1, 4, and 5 is one example). This expression produces **true** or **false** as a result.

a)

b)

c)

d)

e)