Using the operator precedence table above, evaluate each expression and state what gets printed. Remember short-circuit evaluation with && and ||.

```
int a = 2;
int b = -1;
int c = 7;
boolean exp1 = a + b * c < a + c * b; _________ (value of exp1)

boolean exp2 = !(c + a >= b); _________ (value of exp2)

boolean exp3 = b - a != c; _________ (value of exp3)

boolean z = exp1 || exp2 && !exp3;
System.out.println( "z = " + z );
```

```
z = exp1 && exp2 || exp3;
System.out.println( "z = " + z );
```

```
int d = b-- + a++ % 4 * ++c * 2;
System.out.println( "a = " + a );
System.out.println( "b = " + b );
System.out.println( "c = " + c );
System.out.println( "d = " + d );
```

Which of the following are valid Java identifiers? (Circle your answer(s)).

CSE_11  sEvEnTeEn  1stJavaClass  My-First-Java-Class
boolean  Float   My1stJavaClass  CSE11Is#1

In general, you should define instance variables to be _____ (write correct letter here)

A. final
B. public
C. private
D. static

(Continued on other side)
What gets printed with each of the following statements?

```java
int a = 1;
int b = 3;
int c = 5;
System.out.println( a + b + (c + " = ") + a + (b + c) );  
System.out.println( (a + b) + c + " = "+ a + b + c );  
System.out.println( a + (b + c) + " = "+ (a + b) + c );
```

Given the following code segments, what gets printed?

```java
int x = 5, y = 5, z = 5;
if ( x != 5 )
    if ( y <= 7 )
        z = z + 4;
    else
        z = z + 2;
System.out.println( z );
```

Note - No curly braces.

Given the following nested if-else statements, fill in the blanks of the equivalent logic using cascading if-else-if statements using only the boolean variables `sunny` and `rich` and negation and logical AND or logical OR operators. Note the order of what gets printed. This is straight from the textbook.

```java
boolean sunny = ...;
boolean rich = ...;
if ( sunny )
    if ( rich )
        System.out.println( "Outdoor Concert" );
    else
        System.out.println( "Ultimate Frisbee" );
} 
else 
    System.out.println( "Watch TV" );
```