This exam is to be taken by yourself with closed books, closed notes, no electronic devices.
You are allowed one side of an 8.5"x11" sheet of paper handwritten by you.
(Partial) Operator Precedence Table

<table>
<thead>
<tr>
<th>Operators</th>
<th>Associativity</th>
</tr>
</thead>
<tbody>
<tr>
<td>! ++ --</td>
<td>(pre &amp; post inc/dec) right to left</td>
</tr>
<tr>
<td>* / %</td>
<td>left to right</td>
</tr>
<tr>
<td>+ -</td>
<td>left to right</td>
</tr>
<tr>
<td>&lt; &lt;= &gt; &gt;=</td>
<td>left to right</td>
</tr>
<tr>
<td>== !=</td>
<td>left to right</td>
</tr>
<tr>
<td>&amp;&amp;</td>
<td></td>
</tr>
<tr>
<td>=</td>
<td>right to left</td>
</tr>
</tbody>
</table>

Write variable definitions for the following. Only initialize variables if instructed to.

a) A double precision floating point variable named **foo**
   ______________________________________

b) An integer named **bar** initialized to 420
   ______________________________________

c) A variable named **fubar** initialized to true
   ______________________________________

d) If **foo** (above) was a local variable, what would its initial value be?  ________________________

What gets printed?

```java
public class M2
{
    public static void main( String[] args )
    {
        System.out.println( -1  +  2  -  3  *  5  /  4  );  _________
        System.out.println(  6  +  7  %  11  +  9  *  10  );  _________
    }
}
```

What gets printed?

```java
public class M3
{
    public static void main( String[] args )
    {
        int fire = 10, air = 4;
        fire = fire * (air + 7);  
        System.out.println( "fire = " + fire );  

        fire = 6 – air++;  
        System.out.println( "fire = " + fire + "; air = " + air );

        fire = 8;  
        air = --fire + 4;  
        System.out.println( "fire = " + fire + "; air = " + air );
    }
}
```
What gets printed?

```java
public class M5
{
    public static void main( String[] args )
    {
        final int MAX = 2;
        int i = -1, j = -1;

        for (i = 0; i < MAX; ++i )
        {
            for (j = 1; j <= MAX; ++j )
            {
                System.out.println( i + " " + j );
            }
        }
        System.out.println( i + " " + j );
    }
}
```

What gets printed if the value of the actual argument passed to this method is 5?

```java
public void m6( int x )
{
    int y = 0;
    if ( x <= 1 )
        y = 3;
    else if ( x <= 2 )
        y = 5;
    else if ( x == 3 || x >= 4 )
        y = 7;
    else
        y = 9;
    System.out.println( y );
}
```

What gets printed if the value of the actual argument passed to this method is 2?

```java
public void m6( int x )
{
    int y = 0;
    if ( x <= 1 )
        y = 3;
    if ( x <= 2 )
        y = 5;
    if ( x == 3 || x >= 4 )
        y = 7;
    else
        y = 9;
    System.out.println( y );
}
```

What gets printed?

```java
public class M7
{
    public static void main( String[] args )
    {
        int x = 5;
        int y = 8;

        System.out.println( true && !(x <= y) );
        System.out.println( 4 + x < y );
        System.out.println( false || (y >= 8) );
        System.out.println( ++x < y );
    }
}
```
The following is code using a for loop. Write the equivalent using a while loop.

```java
int x = 5;
for ( int i = 0; i < 15; i++ )
{  
  x = x * x;
}
```

Assume the following method is in Picture.java and assume the width and height are evenly divisible by 4.

```java
Pixel[] pixelArray = this.getPixels();
for(int i = pixelArray.length/4; i < (3*pixelArray.length)/4; i++)
{
  pixelArray[i].setColor(Color.BLACK);
}
```

What does the above code do (Pick the BEST answer available): _______

A) This code modifies the top ¼ and bottom ¼ of the picture
B) This code modifies the left ¼ and right ¼ of the picture
C) This code modifies the middle half (from the top and bottom) of the picture
D) This code modifies the middle half (from the left and right) of the picture
E) None of the above

What is stored in the memory location allocated for the variable `foo` for the following: ____

```java
int foo = 42;
```

A) foo  
B) a reference (or address in memory) to where the value 42 is stored  
C) the value 42  
D) int
Why does this code have an error (assume this is in Picture.java)?

```java
Pixel[] pixelArray = this.getPixels();
Pixel q;
Pixel p;

for ( int index = 0; index < pixelArray.length; index++ )
{
    p = pixelArray[index];
    q = pixelArray[index+1];
    p.setRed(q.getRed());
    p.setBlue(q.getRed());
    p.setGreen(q.getRed());
}
```

A) It tries to access `pixelArray[-1]`
B) It tries to access `pixelArray[0]`
C) It tries to access `pixelArray[pixelArray.length]`
D) It tries to access `pixelArray[pixelArray.length+1]`
E) None of the above

What is printed by the following code?

```java
int foo = 42;
int bar = 37;
boolean foobar = (foo == bar);
System.out.println(foobar);
foo = bar;
System.out.println(foobar);
System.out.println(foo == bar);
```

What is stored in the memory location allocated for the variable `foo` for the following: ______

```java
String foo = "42";
```

A) foo  
B) a reference (or address in memory) to where the string "42" is stored  
C) the value 42  
D) String

Assume there is an array properly defined and a variable named `bar` references this array.

Write the expression to access the first element in this array? __________________________

Write the expression to access the last element in this array? __________________________

Which of the following are valid Java identifiers? (Circle your answer(s).)  
[+1 – correct; -1 – incorrect; No negative score]

```
this&that  thisRthat  This_2_That  integer
nine2five  n!ne_2_5  9_2_5  int
```
What order are pixels changed given this code in Picture.java (assuming all variables are correctly defined and set)? Write the letter of the correct order here: _____

```
Pixel p;
for ( int i = 0; i < this.getWidth(); i++ )
{
    for ( int j = 0; j < this.getHeight(); j++ )
    {
        p = getPixel( i, j );
        p.setColor( Color.Black );
    }
}
```

A) L->R; Top->Down  B) L->R; Bottom->Up  C) R->L; Top->Down  D) R->L; Bottom->Up  E) Top->Down; L->R  F) Bottom->Up; L->R

Fill in the blanks so the code will change pixels in the order described by pattern C above.

```
Pixel p;
for ( int __ = _________________________; __ _____ _________________________; _____ )
{
    for ( int __ = _________________________; __ _____ _________________________; _____ )
    {
        p = getPixel( x, y );
        p.setColor( Color.Black );
    }
}
```

If this is the method header for a method definition

```
public int foo( int x, String s ) { /* Method body */ }
```

Which of the following would be a correct way to use this method? ________

A) foo( 42.420, "A" );  
B) double odds = foo( 42.420, "A" );  
C) boolean odds = foo( 42, "A" );  
D) int odds = foo( 420, "A" );  
E) short odds = foo( 420, "A" );
Which is the correct \textbf{if} statement to make this code change the bottom half of a Picture to black? 
\textbf{_____}
You can assume even dimensions.

```java
public void paintItBlack()
{
    Pixel pixel = null;
    for ( int y = 0; y < this.getHeight(); y++ )
    {
        for ( int x = 0; x < this.getWidth(); x++ )
        {
/* MISSING IF STATEMENT */
            pixel = this.getPixel( x, y );
            pixel.setColor( Color.BLACK );
        }
    }
}
```

Which is the correct \textbf{if} statement to make the above code change the left half of a Picture to black? 
\textbf{_____}
You can assume even dimensions.

A) \textbf{if} ( x < this.getHeight() / 2 )
B) \textbf{if} ( y < this.getHeight() / 2 )
C) \textbf{if} ( x < this.getWidth() / 2 )
D) \textbf{if} ( y < this.getWidth() / 2 )
E) \textbf{if} ( x > this.getHeight() / 2 )
F) \textbf{if} ( y > this.getHeight() / 2 )
G) \textbf{if} ( x > this.getWidth() / 2 )
H) \textbf{if} ( y > this.getWidth() / 2 )

Trace the following program and specify its output.

```java
public class Trace
{
    public static void main( String[] args )
    {
        System.out.println( "main" );
        foo1();
        System.out.println( "main" );
        foo2();
        System.out.println( "main" );
    }

    public static void foo2()
    {
        foo3();
        System.out.println( "foo2" );
    }

    public static void foo1()
    {
        System.out.println( "foo1" );
        foo2();
    }

    public static void foo3()
    {
        System.out.println( "foo3" );
    }
}
```