INSTRUCTIONS

This is a closed book and note examination, except for one 8 1/2 x 11 inch paper sheet of notes, both sides. There is no interpersonal or other communication or discussion (except with the prof. or proctors) or sharing of information. Cell phones or any electronic devices (other than a calculator) are strictly forbidden! If you need to leave the room, leave your paper and ask permission. You may take NOTHING with you to leave the room. Failure to abide by any of these terms will result in a grade of zero for the exam.

You must show a picture ID to the instructor or TA when you turn in your exam.

NAME PRINTED CLEARLY: ________________________________

NAME SIGNED: ________________________________

------- For scoring use only. Do not write below this line ---------
Section 1 – Multiple choice and short answer (13 questions, 3 points each)
Select the best answer to the question from the choices provided.

1. When a .java file has javadoc comments (they start with /**, use tags such as as @param, @return, and @author, the comment is just above the method it documents, etc.), what happens when you click "javadoc", which runs the javadoc processing software?
   a. Code to check for mistakes is generated.
   b. The entire .java file is printed.
   c. A html Web page of documentation, the javadoc comments combined with information about the class and methods, is generated and displayed.
   d. The .java file is compiled into a .class file.

2. Here is some Java code. What’s printed when it runs? (In the first response below, 11 was printed at two 1's in a row, not number eleven. Similarly for all the others.)
   ```java
   int Carly;
   int David;
   Carly = 1;
   David = 2;
   Carly = David; //line 5
   David = Carly; //line 6
   System.out.print(Carly);
   System.out.print(David);
   ```
   a. 11
   b. 12
   c. 21
   d. 22

3. What if lines 5 and 6 were interchanged?  
   ```java
   David = Carly;
   Carly = David;
   ```
   a. 11
   b. 12
   c. 21
   d. 22

4. According to the Mad Ph.D. video, explain briefly in English what the code "new Happy_House()" makes the computer do.
   \____________________________________________________________________
   \____________________________________________________________________

Page 2 of 12
5. A 201 beginner coded into the main method:

```java
public static void main(String[] a)
{
    new Picture(FileChooser.pickAFile());
    //more code tried...
}
```

REWRITE this below to make the Pixel gotten using `getPixel( 2, 3 )` be made black with `setColor(java.awt.Color.Black)` and then the Picture be shown with `show( )` Hint: You MUST declare and use a Picture reference variable, with code like `Picture p`

```java
public static void main(String[] a)
{
    new Picture(FileChooser.pickAFile());
}
```

6. Briefly explain WHY a variable like `p` MUST be used in the above problem, in terms of the purpose of the data stored in `p`. You can refer to the Mad Ph.D. video in your explanation.
7. When is a constructor method called automatically?
   a. Before the new operator builds the object.
   b. Immediately after the new operator builds the object.
   c. Some time after the rest of the program starts using the object.

8. A Java array, made by code like `new int[length]` or `new Picture[length]`
   a. Has a length that grows orshrinks as necessary.
   b. Has a length that grows when elements are defined.
   c. Has a length that never changes after the `new` operation is done.

9. When a computer finds the maximum of numbers from an array by processing the elements one at a time, what should it do to decide whether or not to replace the number stored as the maximum found so far?
   a. Compare the previous element to the current element in the 
      ( ... ) of a `while` statement.
   b. Compare the maximum found so far to the current element in the 
      ( ... ) of a `while` statement.
   c. Compare the previous element to the current element in the 
      ( ... ) of an `if` statement.
   d. Compare the maximum found so far to the current element in the 
      ( ... ) of an `if` statement.

10. This professor put code like
    ```java
    boolean atLeft;
    //... somehow atLeft gets a value.
    line 3  if( atLeft == true )
            { /*if part*/ } 
       else
            { /* else part */ }
    ```
    in this exam to make it absolutely clear to beginners that the if part
    should be run by the computer exactly when the value of variable `atLeft`
    is the boolean value `true`, and the else part should be run exactly when
    that value is `false`. This was done in line 3 above.
    Is it OK in Java to simply write "atLeft" instead of "atLeft == true"?
    a. YES
    b. NO

11. Briefly EXPLAIN your YES or NO choice made above:
In Project 5 you made a class that defined the plan (blueprint, plan of the object’s variables and behaviors) for an **Album** object that enabled someone to set up a digital picture album with the capacity he or she chooses, add pictures to it, and, whenever he or she wants, to see all the pictures together in one "result" picture.

Here are some of the operations that you had to program. However, they are listed in an order that is impossible for a computer to perform or which don’t make the desired things happen. For example, A, to see the result Picture, cannot be performed without first making the Picture, which is listed in later step, step B!

A: Call the `explore` method on the final result `Picture` to see it.
B: Construct the final result `Picture`, blank with a given width and height.
C: Copy the added `Pictures` into the final result `Picture`.
D: Calculate the width and the height that the final result `Picture` should have.
E: Make the array that will be used to store the references to the added `Pictures`.

So, ABCDE is an impossible order. There is ONLY ONE ORDER for these steps that makes sense and will satisfy somebody using the program. Write that order (a permutation of A, B, C, D, E):

**WRITE YOUR ANSWER HERE:** ____________________________

In the same **Album** project, the capacity (which is the array's length) and the number of Pictures currently stored are related how?

a. They are the same.
b. When one goes up, the other goes down.
c. The number of Pictures stored is always less than the capacity.
d. The number of Pictures stored is always less than or equal to the capacity.
e. They are unrelated.
(10 points. Meanings of Java statements.)

(5 points) Circle the correct flowchart for Java's while statement
\[
\text{while ( <TEST> ) \{ <BODY STATEMENTS> \}}
\]

(5 points) Below, draw the flowchart for
\[
\text{if( TEST )}
\text{ \{ IF-BODY \}}
\text{else}
\text{ \{ ELSE-BODY \}}
\]
THIS IS A FREE BLANK PAGE FOR EXTRA SPACE, SCRIBBLING, ETC, added so page turns are avoided for the programming problems!
(30 points. Java programming: nested loop and averaging) Your job is to add a method to the Picture class that will apply a certain digital imaging effect to almost the entire image in "this" Picture object, which your method is called on. (That Picture of course is referred to by this.) First you will write a nested loop of code to print out the locations of the Pixels that will be modified.

**Nested Loop: 15 points.** Code in the method body on the next page a doubly nested loop that uses the given System.out.println( ); method call to print, one by one, the coordinates of all the (x, y) locations of Pixels that do not touch the border line of the Picture. For example, if the dimensions of this Picture are width 5 and height 4, your code should print the 6 coordinate pairs below.

Of course, all the inside, non-border touching Pixel locations must be printed no matter what size is the Picture! So the loops MUST make use of the return values of this.getWidth( ) and this.getHeight( ) to control how many times to repeat, and NOT merely repeat the 2 or 3 times OK for the example but wrong for differently sized Pictures.

**Averaging: 15 points.** Add code that changes the red intensity, stored in each Pixel not touching the border, in the following way:

It retrieves, by 5 method calls of the form this.getPixel(some x location, some y location).getRed() the 5 red intensity values from that Pixel together with the red intensities from the four neighboring Pixels above, below, to the right and to the left.

It computes the average of these 5 red intensity numbers.

· Finally, it sets new red intensity of the inner Pixel (with coordinates (x,y) in the diagram) to this computed average with setRed( ).

The illustration makes clear which Pixels are involved in the average.
//This method definition is to be added to the Picture.java
//file that defines G&E's Pictures class. So, within the method
//below, this.getWidth(), this.getPixel(), etc. can and should
//be used.

class Picture {
    // This method definition is to be added to the Picture.java
    // file that defines G&E's Pictures class. So, within the method
    // below, this.getWidth(), this.getPixel(), etc. can and should
    // be used.
    public void blurRedsInside() {
        int xLoc; // We declare xloc and yLoc for your convenience.
        int yLoc; // You must write the code to initialize these
                   // variables, perhaps multiple times, and use them
                   // in the nested loops.

        System.out.println("(" +xLoc+", " +yLoc+" ")");
        // The above makes the computer print
        // ( numberA, numberB )
        // on one line, where the numbers are the
        // current values of the variables xLoc and
        // yLoc

        // It is good to put your code for averaging
        // just below this comment. Hint: DON'T make a
        // loop to compute the average of 5 numbers!

        // It is good to put the ends of the loop
        // bodies near this place.
    }
}
(10 points. Method-based decomposition, making and calling a method with a parameter) Just read this page: The answer goes on the next page. The purpose is to restructure a program by (1) removing repeated code (2) putting with just one copy of the removed code into a new parametrized method and (3) calling that method in two different cases, with different parameters. In one case, a parameter will be constant; in the other, it will be computed.

Somebody put the `sideCopy` method into the `Picture` class. It copies the given `Picture srcP` into "this" `Picture` at the given Y location `int yWhere`. But, the `boolean` parameter `atLeft` determines whether the copy is put at the left or at the right. See the examples below. (`srcP` is the filled in rectangle.)

```java
public void sideCopy(Picture srcP, int yWhere, boolean atLeft)
{
    if( atLeft == true )
    {
        int y = 0;
        while( y < srcP.getHeight() )
        {
            int x = 0;
            while(x < srcP.getWidth())
            {
                Pixel p = this.getPixel(x, /*SEE THE SPOT BELOW and compare!*/
                    y+yWhere);//p locates Pixel TO BE CHANGED!

                p.setColor(srcP.getPixel(x, y).getColor());
                x = x + 1;
            }
            y = y + 1;
        }
    } else
    {
        int y = 0;
        while( y < srcP.getHeight() );
        {
            int x = 0;
            while( x < srcP.getWidth() )
            {
                Pixel p = this.getPixel(x+this.getWidth()-srcP.getWidth(), /*SPOT!*/
                    y+yWhere);

                p.setColor(srcP.getPixel(x, y).getColor());
                x = x + 1;
            }
            y = y + 1;
        }
    }
} //HA! The 2nd nested loop is just like the first, except for one spot!
}//Answer the question on the next page...
```
Demonstrate that you can make the program better by (A) making a new method containing ONLY ONE doubly-nested LOOP, to be called from two different places in copy, with suitable parameters, and (B) replacing the two doubly-nested for loops with two calls to the copy method.

Your answer to (A) goes in the method below:

```java
private void generalCopy(Picture srcP, int xWhere, int yWhere) {
    //YOU code the doubly-nested loop to do the copying!!
    //Hint: Copy a few lines from page 8 with small changes.
}
```

Your answer to (B) goes in the method below:

```java
public void sideCopy(Picture srcP, int xWhere, boolean atLeft) {
    //YOU rewrite the sideCopy method from above so it has no
    //loops but it calls generalCopy from two places with
    //the correct parameter values.
    if( atLeft == true )
    {
        
    }
    else
    {
        
    }
}
```
(10 points) Harder array programming problem.

Create a static (that is, class) method named `rotateLeft`. Its one parameter must be an `int` array (reference): `rotateLeft(int A[])` It should not return a value (so declare its return type `void`), but it should re-arrange the values stored in the array elements.

The leftmost (first) element should be moved to the end of the array, and the other elements should be shifted one position to the left.

For example, if `AA` refers to the length 4 array containing the numbers below,

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>35</td>
<td>9</td>
<td>17</td>
</tr>
</tbody>
</table>

and the method call `rotateLeft(A)` is done, the `rotateLeft` method should return with array `AA` rearranged and containing:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>9</td>
<td>17</td>
<td>13</td>
</tr>
</tbody>
</table>

Hint 1: It must work an ANY length array (not just 4), so it must (1) get the array length with `A.length` and (2) use a loop. Hint 2: Take care that the number stored in `A[0]` is not lost when your code copies the data from `A[1]` into `A[0]`. 