Grades for the final will be posted on the CMS as soon as it is graded, hopefully tonight but perhaps tomorrow. Grades for the course will take a few days more. You can look at your final when you return in the fall. HAVE A NICE SUMMER!

Please submit all requests for regrades for things other than the final BY 9PM TONIGHT. Use the CMS where possible; email Gries otherwise.

You have 2.5 hours to complete the questions in this exam, which are numbered 0..8. Please glance through the whole exam before starting. The exam is worth 100 points.

Question 0 (1 point). Print your name and **net id** at the top of each page. Please make them legible.

Question 1 (12 points). Loops. Array segment b[0..k] is already sorted (in ascending order), but it may contain duplicates. We want an algorithm that will remove the duplicates, as indicated by the following pre- and post-conditions.

Question 0.	_________	(out of 01)
Question 1.	_________	(out of 12)
Question 2.	_________	(out of 12)
Question 3.	_________	(out of 12)
Question 4.	_________	(out of 08)
Question 5.	_________	(out of 12)
Question 6.	_________	(out of 16)
Question 7.	_________	(out of 17)
Question 8.	_________	(out of 10)
Total	___________	(out of 100)

```
        0                                                    k
Pre: b | sorted                                              |

        0                         h                          k
Post:b | sorted, contains original b[0..k] | this part of b   |
       | with duplicates removed           | is unchanged     |
```

For example, for the array b = {1, 2, 2, 2, 4, 4, 4}, execution of the algorithm sets h to 2 (because there are 3 distinct values) and changes array b to this: b = {1, 2, 4, 2, 4, 4, 4}. The first three values in b are the elements of the original array b, without duplicates.

Write one loop (with initialization) that performs this task. You need not declare any variables; assume they are all declared. You **must** use the invariant shown below. You may write your answer on the back of the next page. Read the invariant carefully before proceeding.

```
        0                        h               p              k
inv: b | sorted, contains original b[0..p] | this part of b | this part of b |
       | with duplicates removed           | is unchanged   | is unchanged   |
```

Question 2 (12 points). Arrays and methods. Some math formulas, like that for the determinant of a 2-dimensional rectangular array, involve such an array but with one of its rows and one of its columns deleted. For example, given array A:

```
        1 2 3 4           array A with          1 2 3
    A = 5 6 7 8           row 1 and col 3       9 0 0
        9 0 0 0           deleted is:           0 1 5
        0 1 5 6
```

Write function reduceArray to calculate this modified array according to the given specifications.

```
/** = an array that is the same as array b but with row r and column c deleted.
        (Remember that the first row is row 0 and the first column is column 0.)
        Preconditions:  (1) b is rectangular
                        (2) 0 <= r < b.length
                        (3) 0 <= c < b[0].length
*/
public static int[][] reduceArray(int [][] b, int r, int c) {




}
```

Question 3 (12 points). Matlab. (a) Write a one-line expression that calculates the average of the positive numbers in array v. You may use function *mean*, which computes the mean (i.e. average) of the values in its parameter array.

(b) Write Matlab code to compute the first n terms of the following sequence. You may not use loops, recursion, or functions that calculate factorial or exponentiation.

$$\sum_{k=1}^{n} \frac{-1^{k+1} x^{k+2}}{(2k-1)\,!} = \frac{1*x^3}{1!} + \frac{-1*x^4}{3!} + \frac{1*x^5}{5!} + \ldots$$

Question 4 (8 points). Consider the class definitions given at the bottom of this page.

Execute by hand a call `F.main(null);` on method `main` of class `F`, given below. After each print statement, write to its right the value of the expression that is printed (or write ERR if the statement is illegal). You need not draw objects, etc., but it will help you if you do. Remember: if the first statement of a subclass constructor is not a call on a superclass constructor, Java inserts a call on a superclass constructor.

```
public class F {
   public static void main(String args[]) {
      BB x= new BB(22);
      BB z= new DD();
      DD y= new DD(88);

      /* a */ System.out.println(“a. ” + x.getX());

      /* b */ System.out.println(“b. ” + y.getX());

      /* c */ System.out.println(“c. ” + z.getX());

      /* d */ System.out.println(“d. ” + x.equals(z));

      /* e */ System.out.println(“e. ” + y.getIT());

      /* f */ System.out.println(“f. ” + z.getIT());

      /* g */ System.out.println(“g. ” + y.testIT());

      /* h */ System.out.println(“h. ” + z.testIT());
   }
}
```

```
public class BB {
   private int x;

   public BB() {x= 11;}

   public BB(int x) {
     this.x= x;
   }

   public int getIT() {
     return x + 1;
   }

   public int getX() {
     return x;
   }
}
```

```
public class DD extends BB {
   private static int y;

   public DD() {
      this(0);
   }

   public DD(int y) {
      this.y= y;
   }

   public DD(DD ob) {
      y= ob.y;
   }

   public int getIT() {
      return getX() + y;
   }

   public boolean testIT() {
      return y>0;
   }
}
```

Question 5 (12 points). Strings and recursion.

(a) Write the following function. For example, `occ('a', "abaaab")` is 4, `occ('b', "abaaab")` is 2, and `occ('x', "abaaab")` is 0. You may not use a loop; the only String functions to be used are `charAt` and `substring` (not `indexOf`, for example); and you must use recursion.

```
/** = number of occurrences of c in s */
public static int occ(char c, String s){









}
```

(b) Write the following function. For example, `remDup("abaaabc")` can be `"abc"` or `"bac"` or `"cba"` (depending on which duplicates are removed). You may not use a loop; the only String functions to be used are `charAt` and `substring` (not `indexOf`, for example); and you must use recursion. You may use calls on function `occ`, specified above. Hint: Think of removing `s[0]` if it appears later in `s`.

```
/** = s but with duplicate characters removed –the elements in the result need not
        be in the same order as their first occurrences in s */
public static String remDup(String s) {









}
```

Question 6 (16 points). Classes. Websites like MySpace, Facebook, and Friendster are great, but you have your own great idea for a social networking site. In this question, you will develop two classes for use in running it. *Write specifications for all methods*. This exercise requires writing some calls on methods in class `Vector`. If you are not sure what the `Vector` methods are, do what you think is right and put comments that explain your method calls. Make fields and method private or public as appropriate.

(a) Write a class `Person`. A person has a name, a hometown, and a list of friends (a `Vector` of `Persons`). Your class needs a constructor that allows a user to give values for all three fields, a procedure `addFriend` to add a new friend, and getter methods for the fields. Finally, it should have a constructor with no parameters that initializes the name to “John Doe”, the hometown to “Middle USA”, and with no friends; its method body should have one statement only, a call on the other constructor.

(b) Many “people” on sites like MySpace and Friendster aren't actually people but animals, bands, and other groups. Write a subclass `Band` of Person. Besides having friends, a `Band` has members. So, `Band` has an additional field (a `Vector` of `Persons`) to contain its members, as well as methods `addMember` and `removeMember` to deal with people joining and leaving the band. The constructor of `Band` should allow the fields of the superclass to be set as in the superclass and also needs a parameter that is a vector of members.

Use this page for Question 6.

Question 7 (17 points). Miscellaneous.

(a) Name 3 kinds of layout managers and the component with which each is usually associated.

(b) Define the word "type".

(c) Name the four kinds of variable in Java and indicate where each is declared.

(d) To the right is a definition of a class `Q7d.`

First, draw the file drawer for Q7d just before execution starts.

Second, write down the steps in executing the expression **new** Q7d(3, 4). With each, actually do the step yourself, drawing any objects and calls for frames that are required, putting the objects where they belong (place frames for calls anywhere on the page).

If you have to execute a method call, draw the frame for the call, but you do not have to explain the steps in executing the method call.

```
public class Q7d {
   public static int st= 5;
   public int f;
   public Q7d(int f, int t) {
      this.f= f;
      st= st + t;
   }
}
```

Question 8 (10 points). Algorithms.
Write algorithm `Partition` as a method, complete with method header (giving the parameters, for example) and a suitable specification —a precondition and postcondition. You may give the pre- and postconditions as formulas or as pictures. If you have to swap two variables x and y (say), just write "swap x and y"; you need not write the sequence of three statements to swap them.