Final Exam Spring 2000 (Do 8 of the 10 following problems)

1. Loop Trace
Trace the following code, indicating the output at each loop as shown:

```
public class  iBisect {
// Does show bisection or bracketing

   public static void main (String[] args) {
   // Does find the integer root
      int x,y, hi, lo;
      y = 81;  x = y;
      lo = 0;  hi = y;
      while ( x * x != y ) {
         System.out.println (x);
         if (x * x < y)
            lo = x; // increase the low
         else
            hi = x; // decrease the high
         x = (hi + lo) / 2;  // find mid
      }//end while loop
      System.out.println ("The value is " + x );
   }//end routine main

}//end class Bisect
```

2. Logical Play
Games involving two individuals or teams often have a form similar to the following;
The game ends when either score is over some limit such such as 21, but
the difference in scores must also be larger than some other difference such as 3.
Write the condition for halting , in terms of the two scores, say a and b.
then write the opposite condition for continuing to play.

3. Leap Year
Prove whether the following piece of code indicates that a year is a leap year,
where year is a positive integer, and leap is a boolean.
A leap year is defined as one which is divisible by 4 and
not divisible by 100 unless it is also divisible by 400.
For example, 1984 and 2000 are leap years,
whereas 2001 and 1900 are not leap years.

```
if (year % 400 == 0)
   leap = true;
else
   if (year % 4 == 0)
      if (year % 100 == 0)
         leap = true;
      else
         leap = false;
   else
      leap = false;
```

Package this above piece of code properly as a function.

4. VIN: Vehicle Registration Number
Every car manufactured since 1980 is given a special vehicle registration number called VIN. It is a string of 17 characters with the the following format:

```
0       country
1 - 2  make
3 - 5  model
6       body code
7       restraint system
8       check digit (0..9 and X)
9       year
10      plant
11-16  serial number
```

The country code in position 0 indicates the country of manufacture; for example, '1' is US, '2' is Canada, 'J' is Japan, 'S' is England, 'Z' is Italy, etc.
The model year is given by a single character, in position 9, beginning with 'A' for 1980, 'B' for 1981, 'C' for 1982, etc with the letters I, O, Q, and U skipped (so 'X' is 1999).
Write a function `modelYear(vin)` which returns the year corresponding to the vin.
(note: there is much repetition here; you need not fill in all the details over all the many years; just provide sufficient code to show that you know the way to do it, and could finish if you had more time.)

5. Nice Numbers
Given a number of routines (ready to use) such as:
spellDigit(i), which spells out the first ten digits 0 to 9,
spellTeen (t) which spells out the integers from ten to twenty,
use and reUse these routines to write a method spellBigNum(b)
which spells out any number less than 100.
For example spellBigNum(67) outputs “sixty seven”.

6. Format Money
The following code does output a real value as money with two digits after the decimal point (rounded off).
Package this as a routine in Java with the real value passed in (not read in) and the amount printed out.

```
-- Does format money
Box amount ofType real
Boxes iAmount, dollars, cents ofType int
Outputln "Enter a money amount "
Input amount
Outputln amount -- echo
Set  amount = 100.0 * amount
Set iAmount = RealToInt (amount + 0.5)
Set dollars = iAmount / 100
Set cents   = iAmount % 100
Output "$"
Output dollars
Output "."
If (cents <= 9) then
   Output "0"
EndIf
Output cents
```

7. Cash Till class
A till (money drawer or cash register) is a box used in businesses which has various places to hold some denominations of money and to make change. Suppose that the denominations are given by the number of coins as shown.
Draw such a class as a class diagram, and indicate some methods (worth, remove, etc) involving such an object.
Write a method to determine if two such tills are equal (have the same numbers of all denominations).
Write another method to determine if two such tills are equivalent (have the same amount of money but with different denominations).

```
class Till {
// Does provide cash inventory of coins

// Attributes: coin denominations
   private int pennies;
   private int nickels;
   private int dimes;
   private int quarters;

   public Till (int p, int n, int d, int q) {
      pennies = p;
      nickels = n;
      dimes   = d;
      quarters= q;
   }//end constructor Till
}//end class Till
```

8. Count Sort
Countsort is a way of sorting which indicates the rank of all the values of an array.
For example, the largest value will have a rank of zero, whereas the smallest values will have a rank of n, the size of the array.
This sort is done by looping throught each value and counting the number of items which are larger th equal to) this value. The following method loops through each value and calls bigCount to get the coun

```
public static void countSort (int[] theArray) {
// Sorts theArray of integers, giving a rank of each
   int value;
    int num = theArray.length;
   for (int i = 0; i < num; i++) {
        value = theArray[i];
         System.out.print (value + " ");
          System.out.println (bigCount (theArray, value) );
   }//end for
}//end routine countSort
```

Write the function `bigCount(theArray, theValue)` which returns the number of values in theArray which are larger than or equal to theValue.

9. Total Miles
Two arrays, MilesStart and MilesFinish, indicate the start and finish mileage on many refueling stops of trip. Note that the finishing mileage of one refueling stop is the starting mileage of the next stop. An example follows:

```
MilesStart  = {10000, 10150, 10450, 10650 };
MilesFinish = {10150, 10450, 10650, 11010 };
```

write a method to compute the total distance travelled on a trip.
Replace the inner body ofthis method by another way to compute this.

Tank Class

Tanks containing some fluid, are described by the following class
a. Indicate some (2) other possible methods that could be usefull.
b. Define a method pourFrom(T,a) to pour an amount from T to this tank.
c. What assertion (Check and Bounce) would you expect to find in the pourFrom
d. follows below

```
class Tank {
// Does provide for a tank of some fluid

// Attributes
   private int capacity;
   private int level;

   public Tank (int c) {
      capacity = c;
      level = 0;
   }//end constructor Tank

   public int theLevel() {
   // Does return the level in this tank
   }//end function theLevel

   public void add (int amount) {
   // Does add an amount to this tank
   // Check no overflow
   }//end routine add fluid

   public void sub (int amount) {
   // Does subtract amount from tank
   // Check tank does contain enough
   }//end routine subtract amount

   public void fill () {
   // Does fill up this Tank to capacity
   }//end fill

   public void empty() {
   // Does empty out this tank
   }//end routine empty

   public void pourFrom (Tank s, int amount) {
   //Does pour an amount from tank s to this tank
   // Check
   }//end routine pour an amount from s into this

   public boolean isEmpty () {
   // Does tell if this tank is empty
   }//end routine isEmpty

}//end class Tank
```

d. Code the following as a main program, using the methods of the given clas
and any others that you may wish to assume and use (and create later).
It does produce 5 units of fluid given two tanks of 7 and 4 units each.

```
Fill the smaller one (4 units)
Pour it into the larger one
Fill smaller one again
Pour into larger till filled
(leaving one unit in smaller)
Empty the larger one
Pour the smaller into larger
Fill smaller one (with 4)
Pour smaller into larger
```