Advanced Programming
Exam 1 (100 points)
SOLUTION

Name: ____________________________  Section: __________

This a closed book exam. All vector methods will be given. If you need an additional
method, please ask. Good Luck!!!

Part I - Circle the best answer for the following questions (40%)

1. The most time consuming stage of software development is
   a. coding  b. code review  c. compiling  d. testing and debugging

2. The most expensive component of modern computing is
   a. Software  b. hardware

3. Coding is low priority compared to design
   a. TRUE  b. FALSE

4. In object oriented programming data is
   a. Primary  b. secondary

5. The notion of combining state and methods in an object is called
   a. inheritance  b. encapsulation  c. interface  d. none of the above

6. An object is a class instance
   a. TRUE  b. FALSE

7. A class is a ____________ for objects
   a. instance  b. blue-print  c. method

8. Package is a collection of ____________
   a. related classes  b. related methods  c. related variables

9. The key word "extends" refer to the notion of
   a. class definition  b. inheritance  c. state of an object

10. The relationship "is-a" is
    a. inheritance  b. recursion  c. class containment

11. Which one of the following is not a primitive data type
    a. int  b. bool  c. float  d. char

12. Public members of a class are:
    a. Accessible only by the class members
    b. Accessible by all outside objects

13. Objects are _____ from classes
    a. declared  b. constructed

14. Which one of the following is not a primitive data type
    a. int  b. class  c. char  d. boolean
15. The following is valid code:  final int x = 2;  x = x + 1;
   a. TRUE  
   b. FALSE

16. A difference between private and protected member of a class makes sense when we are dealing with
   a. Defining new members of a class   
   b. using inheritance to create new classes   
   c. when we want to protect private members from changes

17. An array size can be changed at run time
   a. TRUE  
   b. FALSE

18. In the declaration int [] A, B;
   a. only A is an array   
   b. only B is an array   
   c. both A and B are arrays

19. Which one of the following is NOT a proper method name?
   a. goForward()   
   b. jumpUp()   
   c. TurnAround()   
   d. goBack()

20. Class interface
   a. provide all class method details   
   b. provide a quick glance at class methods

---

Part II - WRITE NEW METHODS (30%)
1. Study the class definition of matrix
public class Matrix
{
   public Matrix() { }       // unallocated vectors
   public Matrix(int n, int m); // dimensions given
   public Matrix(int n, int m, final int fillValue);
      // and default fill value
   private int NumRows ;
   private int NumColumns ;
   public int final getRows();
   public int final getColumns();
   public void SetDimensions(int r, int c);
};
   a. Name all accessors
   public int final getRows();
   public int final getColumns();
   name all mutators
   public void SetDimensions(int r, int c);
   b.
c. name all constructors?
   public Matrix() {}       // unallocated vectors
   public Matrix(int n, int m); // dimensions given
   public Matrix(int n, int m, final int fillValue);

d. Show Three different ways of creating a matrix.
   Matrix M1 = new Matrix();
   Matrix M2 = new Matrix(2,3);
   Matrix M3 = new Matrix(2,3,1);

Using the available methods only, write new methods:

e. A boolean method to return true if the matrix is square.

   public boolean isSquare()
   {
      return ( numRows == numColumns);
   }

f. Write a boolean method that returns true if the matrix is symmetric (i.e M[i][j]=M[j][i])
   public  void isSymmetric() {
      for (int I=0; I<numRows; I++)
         for (int j=0; j<numColumns; j++)
            if (mat[I][j] != mat[j][I])  return false;
      return true;
   }

g. Write a complete method that will return the sum of the diagonal elements of a matrix, if
   it is square.
   public int diagonalSum() {
      if (isSquare())
      {
         int sum=0;
         for (int I=0; I<numRows;I++)    sum += mat[I][I];
         return sum;
      }
      else  return -999;
   }
public class PersonDB {
    private Vector L = new Vector();
    public PersonDB() {
        public boolean searchDB(OnePerson P) {
            public boolean moveToBack(OnePerson P) {
                public boolean moveToFront(OnePerson P) {
        }
    }

    a. Complete the methods moveToBack(). Method should return false, if object P is not in the list.

    public boolean moveToBack(OnePerson P) {
        if (searchDB(P)) {
            int I = L.indexOf(P);
            L.remove(I);
            L.addElement(P);
        } else return false;
    }

    b. Complete the methods moveToFront(). Method should return false, if object P is not in the list.

    public boolean moveToFront(OnePerson P) {
        if (searchDB(P)) {
            int I = L.indexOf(P);
            L.remove(I);
            L.insertElementAt(P, 0);
        } else return false;
    }

Part III - TRACE THE CODE (20%)
1. Assume that list is a vector. What is the purpose of the following code
   for (int I = 0; I < list.size() / 2; I++)
       { int temp = list[I];
         list[I] = list[list.size() - I - 1];
         list[list.size() - I - 1] = temp;
     }

   This code reverses an array

   2. Consider the following code
      int I = ???
      if (I < 1) { task1();}
      else if (I > 3)
          { task2();}
else
  { task3(); }

Find value of I such that
a. task1 is executed - I = 0 (anything less than 1)

b. task2 is executed – **anything greater to 3** (eg: 4)

c. task3 is executed - **anything greater or equal to 1 but less or equal to 3** (eg: 3)

**Part IV - FIND SYNTAX/LOGIC ERRORS (10%)**
1. find all syntax errors in the following code. Write the corrections to the right of code.

```java
Int I = 0  // missing ;  I capital
while (I < 1) ;  // no semicolon
  {    if  I > 2     // no ( )
        { I = I + 2 } // no semicolon
    I = I + 1;    // ok

2. Find the unreachable statement(if any) in the following code.
if ( I < 1)
  task1();
else if ( I  < 0) task2(); else  task3();

if I<0 then I<1 and therefore  task2 is never reached
```