I. Refer to the following classes for Questions #1-4. (All problems 2 pts each)

```
public class Address
{
    private String myName;
    private String myStreet;
    private String myCity;
    private String myState;
    private String myZip;

    // constructors would go here

    // accessors
    public String getName( )
    { return myName; }
    public String getStreet( )
    { return myStreet; }
    public String getCity( )
    { return myCity; }
    public String getState( )
    { return myState; }
    public String getZip( )
    { return myZip; }
}

public class Student
{
    private int idNum;
    private double gpa;
    private Address myAddress;

    // constructors would go here

    //accessors
    public Address getAddress( )
    { return myAddress; }
    public int getIdNum( )
    { return idNum; }
    public double getGpa( )
    { return gpa; }
}
```

Also a client method has this declaration: Address[ ] list = **new** Address[100];
Also assume that each of the 100 list items has been assigned an Address reference.

Over

1) Here is a code segment to generate a list of *names only.* 1) _____

```
for (int i = 0; i < list.length; i++)
    < line of code >
```

Which is a correct < *line of code* >?

A) System.out.println(Address[i].getName( ));
B) System.out.println(Address.list[i].getName( ));
C) System.out.println(list[i].Address.getName( )) ;
D) System.out.println(list[i].getName( ));
E) System.out.println(list.getName[i]);

2) The following code segment is to print out a list of addresses (i.e. all data in an Address):

```
for (int i = 0; i < list.length; i++)
{
    < more code >
}
```

Which is a correct replacement for < *more code* >? 2) _____

I. System.out.println(list[i].getName( ));
   System.out.println(list[i].getStreet( ));
   System.out.print(list[i].getCity( ) + " ");
   System.out.print(list[i].getState( ) + " ");
   System.out.println(list[i].getZip( ));
II. System.out.println(list[i]);
III. System.out.println(list[i].Address);

A) I only B) II only C) III only D) I and II only E) I, II, and III

3) A client method has this declaration:

```
Student[ ] allStudents = new Student [NUM_STUDS];
                         // NUM_STUDS is an int constant
```

Which is a correct code segment to generate a list of Student names only? 3) _____

I.
```
Address a;
for (int i = 0; i < NUM_STUDS; i++)
{
    a = allStudents[i].getAddress( );
    System.out.println(a.getName( )) ;
}
```

II.
```
for (int i = 0; i < NUM_STUDS; i++)
System.out.println(allStudents[i].getAddress( ).getName( ));
```

III.
```
for (int i = 0; i < NUM_STUDS; i++)
System.out.println(allStudents[i].getName( ));
```

A) I only B) II only C) III only D) I and II only E) I, II, and III

4) Here is a method that locates the Student with the highest idNum:

```
// Precondition: array s of Student is initialized
// Postcondition: Student with highest  idNum  has been returned
public static Student locate (Student[ ] s)
{
    < method body >
}
```

Which of the following could replace < *method body* > so that the method works as intended? 4) _____

I.
```
int max = s[0].getIdNum( );
for (int i = 1; i < s.length; i++)
    if (s[i].getIdNum( ) > max)
    {
        max = s[i].getIdNum( );
        return s[i];
    }
return s[0];
```

II.
```
int index = 0;
int max = s[0].getIdNum( );
for (int i = 1; i < s.length; i++)
    if (s[i].getIdNum( ) > max)
    {
        max = s[i].getIdNum( );
        index = i ;
    }
return s[index];
```

III.
```
int max = 0;
for (int i =1; i < s.length; i++)
if (s[i].getIdNum( ) > s[max].getIdNum( ))
    max = i;
return s[max];
```

A) I only
B) II only
C) III only
D) I and III only
E) II and III only

II. Refer to the following classes for Questions #5-7.

```
public class Ticket
{
    private String myRow;
    private int mySeat;
    private double myPrice;

    //constructor
    public Ticket (String row, int seat, double price)
    {
        myRow = row;
        mySeat = seat;
        myPrice = price;
    }

    //accessors not coded would be:  getRow( ) , getSeat( ), and getPrice( )
}

public class Transaction
{
    private int myNumTickets;
    private Ticket[ ] tickList;

    //constructor
    public Transaction(int numTicks)
    {
        myNumTickets = numTicks;
        tickList = new Ticket[numTicks];
        String row;
        int seat;
        double price;
        for (int i = 0; i < numTicks; i++)
        {
            < read user input for row, seat, and price >
                . . .
            < more code >  // creates Ticket objects
        }
    }

    //returns total amount paid for this transaction
    public double totalPaid( )
    {
        double total = 0.0;
        < code to calculate amount >
        return total;
    }
}
```

Over

5) Which of the following correctly replaces *< more code >* in the Transaction constructor to initialize the tickList array? 5) _____

A) tickList[i] = **new** Ticket (getRow( ), getSeat( ), getPrice( ));
B) tickList[i] = **new** Ticket(row, seat, price);
C) tickList[i] = **new** tickList(getRow( ), getSeat( ), getPrice( ));
D) tickList[i] = **new** tickList(row, seat, price);
E) tickList[i] = **new** tickList(numTicks);

6) Which represents correct *< code to calculate amount >* in the totalPaid method? 6) _____

A)
```
for (int i = 0; i < myNumTickets; i++)
    total += tickList[i] .myPrice;
```
B)
```
for (int  i = 0; i < myNumTickets; i++)
    total += tickList.getPrice[i] ;
```
C)
```
for (int i = 0;  i < myNumTickets; i++)
    total += tickList[i].getPrice( ) ;
```
D)
```
Transaction t;
for (int i = 0; i < t.numTicks; i++)
    total += t.tickList[i].getPrice( );
```
E)
```
Transaction t;
for (int i = 0; i < t.numTicks; i++)
    total += t.tickList[i].myPrice;
```

7) Suppose it is necessary to keep a list of all ticket transactions. A suitable declaration would be which of the following? 7) _____

A) Transaction[ ] listOfSales = **new** Transaction[NUMSALES];
B) Transaction[ ] listOfSales = **new** Ticket[NUMSALES];
C) Ticket[ ] listOfSales = **new** Transaction[NUMSALES];
D) Ticket[ ] listOfSales = **new** Ticket[NUMSALES];
E) Transaction[ ] Ticket = **new** listOfSales[NUMSALES];

III. Use the following to answer #8) below.
A class of 80 students rated their computer science teacher on a scale of 1 to 10 (1 means awful and 10 means outstanding). The responses array is a 30-element integer array of the student responses. An 11-element array named freq will count the number of occurrences of each response. For example, freq[6] will count the number of students who responded 6. The quantity freq[0] will not be used.

Here is a program that counts the students' responses and outputs the results.

```
public class StudentEvaluations
{
    public static void main(String args[ ])
    {
        int[ ] responses = {6,6,7,8,10,1,5,4,6,7,5,4,3,4,4,9,8,6,7,10,6,7,8,8,9,6,7,8,9,2};
        int[ ] freq = new int[11];
        for (int i = 0; i < responses.length; i++)
            freq[responses[i] ]++;
        System.out.print("rating\tfrequency\n");  // output results, recall that \t is a tab
        for (int rating = 1; rating < freq.length; rating++)
            System.out.print(rating + "\t" + freq[rating] + "\n");
    }
}
```

8) Suppose the last entry in the initializer list for the responses array was incorrectly typed as 12 instead of 2. What would be the result of running the program? 8) _____

A) A rating of 12 would be listed with a frequency of 1 in the output table.
B) A rating of 1 would be listed with a frequency of 12 in the output table.
C) An ArrayIndexOutOfBoundsException would be thrown.
D) A StringIndexOutOfBoundsException would be thrown.
E) A NullPointerException would be thrown.

IV. Consider this class to answer #9) below:

```
public class Book
{
    private String myTitle;
    private String myAuthor;

    //constructor
    public Book(String title, String author)
    {
        myTitle = title;
        myAuthor = author;
    }

    // display title, author
    public void display( )
    {
        implementation code
    }
}
```

A program has this declaration: Book[ ] bookList = **new** Book[SOME_NUMBER];

9) Suppose bookList is initialized so that each Book in the list has a title and author. 9) _____
Which of the following will display the title and author of each book in bookList?

A) **for** (**int** i = 0; i < bookList[i].length; i++)
bookList[i].display( );
B) **for** (**int** i = 0; i < bookList.length; i++)
bookList[i].display( );
C) **for** (**int** i = 0; i < bookList.length; i++)
bookList.display( );
D) **for** (**int** i = 0; i < bookList.length; i++)
Book.display( );
E) **for** (**int** i = 0; i < bookList.length; i++)
Book[i].display( );

V. Consider this class to answer #10) below:

```
public class BingoCard
{
    private int[ ] myCard;

    // default constructor:  creates BingoCard with 24 random digits in the range 1-90
    public BingoCard( )
    { implementation code would go here }

    // display BingoCard
    public void display( )
    { implementation code would go here }
}
```

A program that simulates a bingo game declares an array of BingoCard with NUMPLAYERS elements, where each element represents the card of a different player.

Here is a code segment that creates all the bingo cards in the game and then displays them:

```
< declare array of BingoCard >
for (int i = 0; i < NUMPLAYERS; i++)
    <construct and display each BingoCard >
```

10) Which of the following is a correct replacement for 10) _____
(1) *< declare array of* BingoCard >, and
(2) *< construct and display each* BingoCard > ?

A)
```
(1) int[ ] BingoCard = new BingoCard[NUMPLAYERS];
(2) myCard[i] = new BingoCard( );
    myCard[i].display( );
```

B)
```
(1) BingoCard[ ] players = new BingoCard[NUMPLAYERS];
(2) players[i] = new BingoCard( );
    for (int k = 0; k < 24; k++)
        myCard[k].display( )
```

C)
```
(1) BingoCard[ ] players = new BingoCard[NUMPLAYERS];
(2) BingoCard[i] = new BingoCard( );
    BingoCard [i].display( );
```

D)
```
(1) BingoCard[ ] players = new BingoCard[NUMPLAYERS];
(2) players[i] = new BingoCard( );
    players[i].display( );
```

E)
```
(1) int[ ] players = new BingoCard [NUMPLAYERS];
(2) players[i] = new BingoCard( );
    players[i].display( );
```

key

1 d
2 a
3 d
4 e
5 b
6 c
7 a
8 c
9 b
10 d