Part I - Circle the best answer for the following questions (30 points – 2 pts each)
1. Java is ______________ programming language
   a. procedural       b. object oriented       c. assembly   d. none of the above

2. One of the strongest features of Java programming language is
   a. efficiency      b. platform independence    c. low level access to memory   d. none of the above

3. The class ______________ helps create and initialize objects
   a. Constructor       b. state variables     c. methods       d. none of the above

4. Java arrays can hold primitives as well as references
   a. TRUE   b. FALSE

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 an instance of a class
   a. TRUE     b. FALSE

7. Package is a collection of ___
   a. related classes  b. related methods    c. related variables

8. Which one of the following is not a primitive data type
   a. int    b. String   c. float    d. char     e. None of the above

9. Public members of a class are:
   a. Accessible only by the class members
   b. Accessible by all outside objects
   c. Accessible only by other public members of the class.
   d. None of the above

10. What can you say about the following code segment? int[] Arr = new int(15); Arr[15] = 15;
    a. Assigns the value 15 to the last location of the array     b. throws array index out of bounds error
    c. The Arr declaration should be int[ ] Arr = new int(15);  d. None of the above

11. If A is an array of ints, which of the following statements best describes the effect of A = NULL;
    a. All elements of A are assigned 0     b. Only the first element of A is assigned 0
    c. All memory allocated for array A is freed during garbage collection  c. None of the above

12. The best algorithm to search a sorted list is
    a. linear search            b. binary search     c. Bubble Sort     d. None of the above

13. What statement best describes the following code segment
    int [] A = new int(15); int[] B = A;
a. B is an exact copy of A  b. B and A refer to the same array  c. B is an illegal declaration  d. none of the above

14. Java throws ________ to recover from runtime errors
a. Compiler messages            b. Compiler warnings     c. Exceptions    d. None of the above

15. Assume you have a machine that can perform million comparisons for a second. If you run linear search algorithm on this machine on a data set of billion records, how long will it take(approximately) to find or not find the entry you are looking for?
   a. 1 second           b. 10 seconds    c. 16 minutes    d. none of the above

Part II – COMPLEXITY OF ALGORITHMS [20 pts]

2. Consider the following code [8 pts]
   
   ```java
   for (int i=0; i < n ; i += 2)
       for (int j = 0 ; j < n/2; j++)
           Statement1;
   ```

   How many times statement1 will be executed? Give the answer in Big O.

3. [12 pts] Consider the following problems. Give the complexity of an algorithm in big O notation. When necessary, briefly describe the algorithm. Give the answer using one of the following. O(1), O(log n), O(n), O(n log n), O(n²), O(n!)
   
   a.  Finding maximum element in an unsorted array
   
   b.  Finding an element in a sorted array using binary search
   
   c.  Reversing an array of integers
d. Inserting an element into the beginning of an array

e. Finding all duplicates in an array

f. Finding the longest increasing subsequence in an array.

Part III – Linked Lists and their Applications [ 30 pts]

(1) (a) List two advantages of using Linked Lists over Arrays
   (i)
   (ii)

(b) In each of the following cases, state whether Array or Linked list is the correct data structure to use.
   (i) The size of the data file is known in advance and many elements needs to be randomly accessed
       - _______________________

   (ii) A List of bank records must be maintained and accessed in many different orders (i.e. sorted by name, age, account balance etc.)
       - _______________________

The following function is supposed to return the length of a singly linked list. The code may or may not have errors. Identify errors (if any) and fix ’em. Rewrite the method in the box provided.

```java
public int length(Node Head) {
    int size = 1;
    Node cur;
    while(cur != NULL){
        size++;
        cur = cur.next;
    }
}
```
(3) Write a method `insertNextTo(Node N, Node M)` that inserts Node N next to Node M. You may assume the following node class.

```java
class Node {
    public int data;
    public Node next;
}
```

For example if the current list is

```
2 4 3 5 null
```

after calling `insertNextTo(seven, three)` we have the list:

```
2 4 3 7 5 null
```

```java
public void insertNextTo(Node N, Node M) {
`
(4) Write a MyLinkedList class method, getNode(int n) that returns the n-th node in the list. You must consider special cases such as n is larger than the size of the list, empty list etc. In all cases where a node cannot be found, return null.

```java
public Node getNode(int n) {
```
PART IV – Strings [ 10 pts]

1. Class Grid has three fields

protected static final int n;    // width of the grid
protected static final int m;    // width of the grid
protected int[][] g = new int[m][n]; // actual board

g contains either 1’s (if the sum of the indices is odd) or 0’s (if the sum of the
indices is even). For example g[0][1]=1 and g[0][0]=0 etc. 0 is considered even.

Write an easy to understand toString() method to represent the key information in the class as a string

public String toString() {

}
PART V - Java IO [10 pts]

1. To read a line from the screen, we use

```java
try {
    BufferedReader is = new BufferedReader(new InputStreamReader(System.in));
    String inputLine;
    while ((inputLine = is.readLine()) != null) {
        System.out.println(inputLine);
    }
    is.close();
} catch (IOException e) {
    System.out.println("IOException: " + e);
}
```

Why do we need to wrap one class in another `BufferedReader is = new BufferedReader(new InputStreamReader(System.in));`?

EXTRA CREDIT PROBLEM (10 points)

Write a method `equivalent(MyLinkedList L)` that returns true if this list is similar to list `L`. Two lists are equivalent if one is a permutation of the other. You may assume both lists have distinct elements. You can also use `compareTo` to see if two nodes are equal. For example: 1 2 3 4 null and 4 2 1 3 are equivalent lists. You must consider all special cases. No partial credit for this problem.

```java
public boolean equivalent(MyLinkedList L) {
```