Special Assignment for Midterm

******** These questions are based on previous midterm questions of several semesters. If you complete this kind of questions, I believe that you will perform well in your Mid and Final Examination. InshaAllah!! *******

Task 1

Write a Java method which takes two integer arrays as parameters. Then, create a new array which has the capacity to store both arrays. Append the given arrays and store them in the new array, created in the method. The method returns the resulting array. For example-

if given parameters are,

array1 = {1,2,3,4} and array2 = {5,6,7,8},

return a new array such as {1,2,3,4,5,6,7,8}.

```
int [ ] appendArrays (int [ ]array1, int [ ]array2) {
// your code…
}
```

Your task is:

i) **Write** the java code

ii) **Create two arrays** in your main method:

X = Last4digitOfYourID % 40

Array1 = [X, X+10, X-100, X+20]

Array2 = [X+20, X+30, X+60, X+40]

iii) **Show** the resulting array.

iv) **Trace the code** for your defined array.

Task 2

Write a method which takes a circular array of integers as parameter, along with its start index and size. The method traverses the array backwards and prints only the even numbers (except ZERO) in the array. For example-

If the array is,

cir = {7,8,0,0,0,1,2,3,4,5,6} with start = 5 & size = 8,

output: 8 6 4 2

```
void print (int []cir, int start, int size) {
}
```

Your task is:

i) **Write** the java code

ii) **Create the array** in your main method:

X = Last4digitOfYourID % 35

cir = {X+7, X+8, 0, 0, 0, X+1, X+2, X+3, X+4, X+5, X+6}

iii) **Show** the output.

iv) **Trace the code** for your defined array

Task 3

Consider the following code on Single linked list-

```
class Node {
    int num;
    Node next;
    public Node(int n, Node x){
        num =n;
        next =x;
    }
}
```

```
public class Quiz {

    public static void main(String[] args) {

        LinkedList ll =new LinkedList();

        ll.insert(X+100);
        ll.insert(X+70);

        ll.remove();
        ll.insert(X+58);
        ll.insert(X+79);

        ll.remove();
        ll.insert(X+21);
        ll.insert(X+11);    }

}
```

```
class LinkedList {
    Node head;
    public LinkedList(){
        head = null;
    }

    void insert(int val){
        Node node = new Node(val,null);
        node.next = head;
        head = node;
    }

    int remove() {
        int val = 0;
        if(head != null) {
            Node prev = null;
            Node cur = head;
            while(cur.next!=null){
                prev = cur;
                cur = cur.next;
            }
            val = cur.num;
            if(cur == head)
                head = null;
            else
                prev.next = null;
        }

        return val;
    }
}
```

Draw the block diagram of the final linked list,

where X = Last4digitOfYourID % 37

Task 4

[DO NOT WRITE CODE/ PSEUDO CODE –Simulate the process and show calculations]
Convert the following in-fix expression to the post-fix notation using stack data structure.
[((a+u-2*(u/2-a))%2-1)*(x+c%u)]%2
Convert the calculated post-fix expression back to an infix expression using stack data structure to evaluate that your first conversion was correct.

Task 5

[Linked List]

Run the Tester class and **DRAW** the resultant list clearly indicating the **head** and the **links** where,
X = Last3digitOfYourID%37

Consider the following code,

```
public class Node {
   int x;
   Node prev;
   Node next;
     public Node (int i, Node p, Node n) {
           x = i;
           prev = p;
           next = n;
     }
}
```

```
public class Tester {
      public staic void main(String[]args) {
           int a [] = {X+1,X+2,X+3,X+4};
           int b [] = {X+5,X+6,X+7,X+8};
           MyList m = new MyList();
           m.myMethod(a,b);
      }
}
```

```
public class MyList{
  public Node head;
    public void myMethod(int [] a, int [] b){
      head = new Node(a[0]* b[0], null,null);
      Node h = head; Node p;
        for(int i = 0;i<a.length;i++){
          p = new Node(a[i],null,null);
          h.next = p;
          h = h.next;
      }
     h.next = head;
     h=head;
       for(int i = 0;i<b.length;i++){
          p = new Node(b[i]-
          a[i],null,null);
          h.prev = p;
          h = h.prev;
    }
  }
}
```

~~Evaluate the value of below expression using stack. Don't just write the answer. Show the steps.~~
~~(1+8/2^2)>=3||4>5&&(7!=6)~~

Task 6

(A+B/C^D)>=E||F>G&&[H!=I]

1) **WHAT TYPE OF EXPRESSION IS THIS**?
2) **Convert** this to postfix expression.

   Where, say operator priority is given in below order

   1. ^
   2. *,/,%
   3.+,-
   4.<,<=,>,>=
   5.==, !=
   6. &&
   7. ||

Task 7

Evaluate the value of below expression using stack. Don't just write the answer. Show each steps.
(1+8/2^2)>=3||4>5&&(7!=6)