3.

```
import java.util.*;
public class Main
{
	public static void main(String[] args) {
	Scanner sc = new Scanner(System.in);
	String s = sc.next();
	int n = sc.nextInt();
	boolean ans = false;
	String sub="";
	String st[] = new String[n];
	sc.nextLine();
	for(int i=0;i<n;i++){
	   st[i]=sc.nextLine();
	}
	StringBuilder sb = new StringBuilder(s);
	String out ="";
	for(String word:st){
	   char num = word.charAt(word.length()-1);
	   int t = num-'0';
	   if(word.charAt(0)=='L'){

	        sb.append(sb.substring(0,t));
	        sb.delete(0,t);
	   }
	   else{
	      sb.insert(0,sb.substring(sb.length()-t));
	       sb.delete(sb.length() - t, sb.length());
	   }

	  String res = new String(sb);
	  out+=res.charAt(0);

	}
	System.out.println(out);

	char[] anagram = out.toCharArray();
	        Arrays.sort(anagram);
	        for(int i=0;i<=s.length()-n;i++){
	            if(i==s.length()-n)
	                sub=s.substring(i);
	            else
	                sub=sb.substring(i,i+n);
```

```
    char[] subcheck=sub.toCharArray();
                        Arrays.sort(subcheck);

                        ans=Arrays.equals(anagram,subcheck);

                        if(ans){
                            System.out.println("Yes");
                    break;
                        }
                    }
                    if(!ans){
                        System.out.println("No");
                    }
            }

}
```

1.

```
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        int n = s.nextInt();
        int k = s.nextInt();
        int[] arr = new int[n];
        for(int i=0;i<n;i++){
            arr[i]=s.nextInt();
        }
        int ans = find(arr,0,n,k);
        System.out.println(ans);
    }
    public static int find(int[] arr,int i,int n,int k){
        int ans1=0,ans2=0;
        if(i>=n) return 0;
        else{
            ans1 = arr[i]+find(arr,i+k+1,n,k);
            ans2 = find(arr,i+1,n,k);
        }
        return Math.max(ans1,ans2);
    }
}
```

2.

```
import java.util.*;
public class Main
{
  public static Set < String > getPermutation (String str)
  {
        Set < String > permutations = new HashSet < String > ();
        if (str == null)
         {
                return null;
         }
        else if (str.length () == 0)
         {
                permutations.add ("");
                return permutations;
         }
        char first = str.charAt (0);
        String sub = str.substring (1);
        Set < String > words = getPermutation (sub);
  for (String strNew:words)
         {
                for (int i = 0; i <= strNew.length (); i++)
                 {
                        permutations.add (strNew.substring (0, i) + first +
                                                        strNew.substring (i));
                 }
         }
        return permutations;
  }

  public static void main (String[]args)
  {
        Scanner sc = new Scanner (System.in);
        String data = sc.next ();
        Set < String > permutations = getPermutation (data);
        int d = sc.nextInt ();
        TreeSet<String> check = new TreeSet ();
        check.addAll (permutations);
        int flag=0;
  for (String e:check)
         {
                if(Integer.parseInt(e)%d==0){
                   flag=1;
                   if(e.charAt(0)=='0'){
                      String sub=e.substring(1);
```

```
                    System.out.println(sub);
                }
                else{
                    System.out.println(e);
                }
                break;
            }
        }
        if(flag==0)
          System.out.println(-1);

  }
}
```