Q60. Code-5 Mark

Madhav is a goverment employee and he is given a task that he need to visit N houses in the area. But he is confused that he has to visit which houses

```
import java.util.Scanner;


public class madhavhouse{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int N = sc.nextInt();
        int M = sc.nextInt();
        int count = 1;
        for (int i = 1; count <=N; i++) {
            int ho = 3 * i + 2;

            if (ho % M != 0) {
                System.out.println(ho);
                count++;
            }
        }
    }
}
```

Q61) You are given a string containing only '1' and '0'. You can delete either the "10" or "00" substring from the string, and after deletion, the remaining string gets concatenated.

```
import java.util.Scanner;
```

```
public class zeronedeletion {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int T = sc.nextInt();

        sc.nextLine();

        for (int t = 0; t < T; t++) {

            String s = sc.nextLine();

            s = s.replace("10", "");

            s = s.replace("00", "");


            System.out.println(s.length());

        }



    }

}
```

Q63)you are given 2 numbers as N1,N2.write a program to calculate the smallest number which is divisible by these two numbers.

```
import java.util.*;

public class seventeenth {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int n = sc.nextInt();

        int m = sc.nextInt();

        for(int i = 1 ; i<=n*m;i++){

            if(i%n==0 && i%m==0){

                System.out.println(i);
                break;
            }
        }
    }
}
```

Q65-66) just use Arrays.sort

Q70) n your Java programming task, you're tackling a challenge related to string manipulation. Your goal is to analyze a given string and generate a list of all possible subsequences from it. A subsequence, in this context, refers to a string that can be formed by removing certain characters from the original string while preserving their relative order. Your task involves designing a function that efficiently generates and collects these subsequences for further processing. Write a java program for the same

```
import java.util.*;
public class Q70 {
    public static void solve(char str[], int n)
    {

        for (int i = 1; i <= n; i++)
        {
            for (int j = 0; j <= n - i; j++)
            {
                int k = j + i - 1;
                for (int l = j; l <= k; l++)
                {
                    System.out.print(str[l]);
                }

                System.out.println();
            }
        }
     }
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        String s = sc.nextLine();
        char[] ch = s.toCharArray();
        solve(ch, s.length());
    }
}
```

Q71)Funny Digits

```
import java.util.*;


public class funnynumber {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int t = sc.nextInt();


        for (int i = 0; i < t; i++) {
            int n = sc.nextInt();
            int result = sol(n);
            System.out.println(result);
        }
    }
    public static int sol(int n) {
        while (!isValid(n)) {
            n--;
        }
        return n;
    }
    public static boolean isValid(int num) {
        String str = Integer.toString(num);
        for (int i = 1; i < str.length(); i++) {
            if (str.charAt(i) < str.charAt(i - 1)) {
                return false;
            }
        }
        return true;
    }
}
```

Q61-just sort

Q62) Exam grader

```
import java.util.Scanner;


public class ExamGrader {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        char[] keys = {'D', 'B', 'D', 'C', 'C', 'D', 'A', 'E', 'A', 'D'};
        int numberOfStudents = scanner.nextInt();


        // 2D array to store students' answers
        char[][] answers = new char[numberOfStudents][keys.length];
        // Input each student's answers
        for (int i = 0; i < numberOfStudents; i++) {
            System.out.println("Enter answers for student " + (i + 1) + ":");
            for (int j = 0; j < keys.length; j++) {
                answers[i][j] = scanner.next().charAt(0);
            }
        }
        for (int i = 0; i < numberOfStudents; i++) {
            int correctCount = gradeExam(answers[i], keys);
            System.out.println(correctCount);
        }
    }
    public static int gradeExam(char[] studentAnswers, char[] correctAnswers) {
        int correctCount = 0;
        for (int i = 0; i < correctAnswers.length; i++) {
            if (studentAnswers[i] == correctAnswers[i]) {
                correctCount++;
            }
        }
        return correctCount;
    }
}
```

Q63)String rotation

```
import java.util.Scanner;
```

```
public class stringrotation {
    public static String reverse(String str) {
        StringBuilder reversed = new StringBuilder(str);
        return reversed.reverse().toString();
    }

    public static String rightRotate(String s, int d) {
        int n = s.length();
        if (n < d) {
            return "-1";
        }
        return s.substring(d) + s.substring(0, d);
    }


    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        String inputString = scanner.next();
        int d = scanner.nextInt();

        String rightRotated = rightRotate(inputString, d);
        String reversedString = reverse(rightRotated);

        System.out.println(rightRotated);
        System.out.println(reversedString);
        scanner.close();
    }
}
```

Q64)mark equal

```
import java.util.Scanner;


public class equalbydoubleortriple {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int n = scanner.nextInt();
        int[] arr = new int[n];
        for (int i = 0; i < n; i++) {
            arr[i] = scanner.nextInt();
        }
        boolean result = canMakeEqual(arr);
        if (result) {
            System.out.println("YES");
        } else {
            System.out.println("NO");
        }
    }
    public static boolean canMakeEqual(int[] arr) {
        for (int i = 0; i < arr.length; i++) {
            while (arr[i] % 2 == 0) {
                arr[i] /= 2;
            }
            while (arr[i] % 3 == 0) {
                arr[i] /= 3;
            }
        }
        for (int i = 1; i < arr.length; i++) {
            if (arr[i] != arr[0]) {
                return false;
            }
        }


        return true;
    }
}
```

Q66)Lucky Number

```
import java.util.Scanner;


public class luckynumber {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);


        // Input the number
        System.out.println("Enter a number:");
        int number = scanner.nextInt();
        // Call the function to check if it's a lucky number
        boolean isLucky = isLuckyNumber(number);


        // Output the result
        if (isLucky) {
            System.out.println(number + " is a lucky number.");
        } else {
            System.out.println(number + " is not a lucky number.");
        }


        // Close the scanner
        scanner.close();
    }


    // Function to check if a number is a lucky number
    public static boolean isLuckyNumber(int num) {
        int originalNumber = num;
        int sum = 0;
        int order = orderOfNumber(num);


        while (num > 0) {
            int digit = num % 10;
            sum += Math.pow(digit, order);
            num /= 10;
        }

        return sum == originalNumber;
    }

    // Function to calculate the order (number of digits) of a number
    public static int orderOfNumber(int num) {
        int count = 0;
        while (num > 0) {
            count++;
            num /= 10;
        }
        return count;
    }
}
```

Q67) Character Insertion in String.
Use replace method of string

Q68)Govinda Number

```
import java.util.Scanner;

public class govindanumber {
    public static void main(String[] args) {
```

```java
        Scanner scanner = new Scanner(System.in);
        int n = scanner.nextInt();
        int result = isGovindaNumber(n);
        System.out.println(result);
    }


    // Function to check if a number is a Govinda number
    public static int isGovindaNumber(int num) {
        // Factorize the number and calculate the sum of digits
        int sumOfDigits = sumOfDigits(num);
        int sumOfPrimeFactors = sumOfDigitsOfPrimeFactors(num);


        // Check if the sums are equal
        return (sumOfDigits == sumOfPrimeFactors) ? 1 : 0;
    }


    // Function to calculate the sum of digits of a number
    public static int sumOfDigits(int num) {
        int sum = 0;
        while (num > 0) {
            sum += num % 10;
            num /= 10;
        }
        return sum;
    }


    // Function to calculate the sum of digits of prime factors of a number
    public static int sumOfDigitsOfPrimeFactors(int num) {
        int sum = 0;
        for (int i = 2; i <= num; i++) {
            while (num % i == 0) {
                sum += sumOfDigits(i);
                num /= i;
            }
        }
        return sum;
    }
}
```

Q69)coins-plz send me the ans
Q70)contact book

```
import java.util.*;
public class binarysearchcontact{
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        sc.nextLine();
        String [] name = new String[n];
        String [] phonenumber = new String[n];
        String [] email = new String[n];


        for(int i = 0 ; i< n;i++){
            name[i]=sc.next();
            phonenumber[i]=sc.next();
            email[i]=sc.next();
             }


             String res = sc.next();
             int result = search(name, res);
             if(result!=-1){
                System.out.println(name[result]);
                System.out.println(phonenumber[result]);
                System.out.println(email[result]);
             }
             else{
                System.out.println("-1");
             }
    }
    static int search(String[] names,String target){
        for(int i =0 ; i<names.length;i++){
            if(names[i].equals(target)){
                return i;
            }
        }
        return -1;
    }
}
```

Qu71)song search

```
import java.util.*;


public class binarysaearchsong {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        sc.nextLine(); // Consume the newline
        String[] arr = sc.nextLine().split(" ");
        Arrays.sort(arr);
        String target = sc.nextLine();
        int result = binarySearch(arr, target);
        System.out.println(result);
    }
    public static int binarySearch(String[] arr, String target) {
        int left = 0, right = arr.length - 1;
        while (left <= right) {
            int mid = left + (right - left) / 2;
            int cmp = target.compareTo(arr[mid]);

            if (cmp == 0) {
                return mid;
            } else if (cmp < 0) {
                right = mid - 1;
            } else {
                left = mid + 1;
            }
        }
        return -1;
    }
}
```