

Q) Q/p of :

class Testclass {

public static void main (String [] args) {

int count = 8;

count = count ^ 1;

System.out.println(count);

}

11 marks

Aus: 9

Q) Which of the following is not a primitive data type

Aus: String

Q) Which of the following data types is used to store floating-point numbers with greater precision.

Aus: double

class Testclass {

public static void main (String [] args) {

int a = 10;

int b = 3;

System.out.println(a/b);

}

J.

Aus: 3

Q) import java.util.*;

class TestoperatorExamp {

public static void main (String [] args) {

int a = 5, b = 10;

int result = (a > b) ? a : b;

System.out.println(result);

}

Aus: 10;

⑥ class OperatorProcedure example {
public static void main (String [] args) {
int a = 3, b = 2; // data - value
int result = a + b + c; // calculate
System.out.println (result);
}
}

Ans: 11

⑦ class TestClass {

public static void main (String [] args) {

int x = 5;

int X = 10;

int sum = x + X;

int bitwiseResult = x | X;

System.out.println (sum);

System.out.println (bitwiseResult);

}

3.

Ans: 15
15

⑧ class Division {

public static void main (String [] args) {

double num1 = 10.5;

double num2 = 3;

int result = (int) (num1 / num2);

System.out.println (result);

}

3.

Ans: 3

9) class complex {
 public static void main (String [] args) {
 int a = 5, b = 2, c = 3, d = 4;
 int result = a + b * c / d - b;
 System.out.println(result);
 }
}

Ans: 4

10) class demo {
 public static void main (String [] args) {
 String text = "Hello, World!";
 System.out.println(text);
 }
}

Ans: "Hello, World!"

11) Which of the following data types is used to store single character.

Ans: char

12) import java.util.*;
class SelectionExample {
 public static void main (String [] args) {
 int x = 8, y = 4;
 boolean result = (x != y);
 System.out.println(x != y);
 System.out.println(result);
 }
}

Ans: true.

(13) class arithmetic {
public static void main (String [] args) {
 char ch = 'A';
 System.out.println (ch);
}}

3. Ans: A.

(14) class DataTypesMCQ {
public static void main (String [] args) {
 int a = 10;
 double b = 5;
 System.out.println (a/b);
}

3.
Ans: 2.0

(15) class Testclass {
public static void main (String [] args) {
 int a = 5;
 int b = 10;
 int sum = a+b;
 int bitwiseAnd = a&b;
 int bitwiseOr = a|b;
 System.out.println (sum);
 System.out.println (bitwiseAnd);
 System.out.println (bitwiseOr);
}

3. Ans: 15
 0
 15

WEEK-1 - SKILL BUILDER

- ① Eltonia is responsible for monitoring the performance of 2 reactors in a factory. Assume Eltonia is displaying the reactor's temperature which closer to 100, ad the difference from 100.

Code:

```

import java.util.Scanner;
class Main {
    public class static void main (String [ ] args) {
        int m,n,d1,d2;
        Scanner scan = new Scanner (System.in);
        m = scan.nextInt();
        n = scan.nextInt();
        if (m > 100) {
            d1 = m - 100;
        } else {
            d1 = 100 - m;
        }
        if (n > 100) {
            d2 = n - 100;
        } else {
            d2 = 100 - n;
        }
        if (d1 > d2) {
            System.out.println("The integer closer to 100 is " + n +
                " with a diff of " + d2);
        } else {
            System.out.println("The integer closer to 100 is " + m +
                " with a diff of " + d1);
        }
    }
}

```

Input: 90

80

Output: The integer closer to 100 is 90 with a difference

100 is 90 with a difference

- ③ Dave got 2 students who wants to help with their doubts. Each student gives an integer and wants to find if one integer positive while the other is not divisible by 3.

Program:

```
import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        Scanner scan = new Scanner (System.in);
        int n1, n2;
        n1 = scan.nextInt();
        n2 = scan.nextInt();
        if ((n1 >= 0 && n2 <= 3) || (n2 >= 0 && n1 % 3 != 0)) {
            System.out.println ("One of the integers is positive and other
                is not divisible by 3");
        } else {
            System.out.println ("Neither of the integers meet the
                conditions");
        }
    }
}
```

Input: 4
3

Output: One of the integers is positive and other is not divisible by 3.

- ④ Create a program that takes an integer input and converts it into a double prints both the integer and double.

Program:

```
import java.util.Scanner;
class Main {
    Scanner scan = new Scanner (System.in);
    int n = scan.nextInt();
    System.out.println ("Original Integer: " + n);
    System.out.println ("Converted double: " + (double)n);
}
```

Input:

20

Output:

Original integer: 20

Converted Double: 20.0

④

User gives User 2 integers. The task is to check if the sum of these 2 numbers is a multiple of their product.

Program:

```

import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        int m,n;
        Scanner scan = new Scanner (System.in);
        m = scan.nextInt();
        n = scan.nextInt();
        if ((m+n) % (m*n) == 0)
            System.out.println ("Sum is Multiple of Product");
        else
            System.out.println ("Sum is not multiple of product");
    }
}

```

Input:

1 2

Output:

Sum is Not Multiple of Product

- ⑤ write a program that can take the radius as input and displays the circumference and area of the circle.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main (String [] args){  
        final double pi = 3.14159;  
        Scanner scan = new Scanner (System.in);  
        double r = scan.nextDouble();  
        double c = 2 * pi * r;  
        double a = pi * r * r;  
        System.out.printf ("Circumference : %.2f meter", c);  
        System.out.print ("Area : %.2f square meter", a);  
    }  
}
```

Input:

3.0

Output:

Circumference : 18.85 meter

Area : ~~28.27~~ 28.27 square meter

- ⑥ write a program that takes an integer input as the number of bits N as input. The outputs are the value of the lowest N bits of the integer.

Input: 85
2

Output: Result : 1

Program:

```

import java.util.Scanner;
class Main {
    public static void main(String[] args) {
        int m, n;
        Scanner scan = new Scanner(System.in);
        m = scan.nextInt();
        n = scan.nextInt();
        int mark = (1 << n) - 1;
        int r = m & mark;
        System.out.print("Result: " + r);
    }
}

```

3.

- ② Write a program to check whether one of the integers is both less than or equal to zero and odd.

Program:

```

import java.util.Scanner;
class Main {
    public static void main(String[] args) {
        int n, m;
        Scanner scan = new Scanner(System.in);
        n = scan.nextInt();
        m = scan.nextInt();
        if((m <= 0 & m % 2 != 0) || (n <= 0 & n % 2 != 0)) {
            System.out.println(true);
        } else {
            System.out.println(false);
        }
    }
}

```

Input:

-45
10

Output: true

Q ⑧ Write a program to determine if the daily average expense is greater than two of the categories. Specifically all 3 add check if it is greater than any 2.

Program :

```
import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        int a, b, c;
        Scanner scan = new Scanner (System.in);
        a = scan.nextInt();
        b = scan.nextInt();
        c = scan.nextInt();
        if ((a > args) & (b > args))
            System.out.println ("Avg is greater than " + a + " and " + b);
        else if (a < args & b < args)
            System.out.println ("Avg is greater than " + a + " and " + c);
        else
            System.out.println ("Avg is not greater than both any 2 expenses");
    }
}
```

Input :

⑥

```
5  
100  
10
```

Output :

38.33

Average is greater than both 5 and 10.

- ① Maria - a software engineer, working on program. Write a program that takes 2 integers as input and checks if both the integers are either odd or even.

Program:

```
import java.util.Scanner;
class Main {
    public static void main(String[] args) {
        int m, n;
        Scanner sc = new Scanner(System.in);
        m = sc.nextInt();
        n = sc.nextInt();
        if ((m % 2 == 0 && n % 2 == 0) || (m % 2 != 0 && n % 2 != 0)) {
            System.out.println("Both numbers are either even or odd");
        } else {
            System.out.println("Both numbers have different parities");
        }
    }
}
```

Input :

2

-4

Output

Both numbers are either even or odd.

- ② Write a program to help mickey compute the result of the bitwise AND operation of the given stone number with 15 and print the result.

Program

```
import java.util.Scanner;

class Main {
    public static void main (String [] args) {
        int n;
        Scanner sc = new Scanner (System.in);
        n = sc.nextInt();
        System.out.println (n % 15);
    }
}
```

Input 1 :

25

Output 1 :

9

3

Input 2 :

9

2

Output 2 : 9

0

Input 3 :

212

Output 3 :

4

- ① The ruler of Arithmetica, King Taker needs help to unlock castle. The lock is encoded with a number and the calculator will only open if you apply a right shift by 2 on the no.

Program:

```
import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        int n;
        Scanner sc = new Scanner (System.in);
        n = sc.nextInt();
        n = n >> 2;
        System.out.println (n);
    }
}
```

Input :

16

Output

4

Input :

4

Output

1

②

- Mandy is working on a cybersecurity project that involves basic encryption techniques. She wants to write a program that takes an integer number or performs a bitwise XOR to flip all bits.

Program :

```

import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        int n;
        Scanner scan = new Scanner (System.in);
        n = scan.nextInt();
        System.out.println ("Result : " + (n * n));
    }
}

```

Input : 0 Output : 255

- ③ Write a program to implement the logic to check if the provided integer is the square of an Integer and return result.

Program :

```

import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        int n;
        double m;
        Scanner sc = new Scanner (System.in);
        n = sc.nextInt();
        m = Math.sqrt (n);
        boolean result = (m * m == n) ? true : false;
        System.out.println ("Result : " + result);
    }
}

```

Input : -90

Output : False

Input : 16

Output : True.

WEEK - 2 - MCQ

① class Looptest {
 public static void main (String [J] args) {
 int i = 1;
 do {
 System.out.print(i + " ");
 i += 2;
 } while (i <= 8);
 } } Ans : 1 2 4 6 8

② class Test {
 public static void main (String [J], args) {
 int i = 10;
 do {
 System.out.print(i + " ");
 i -= 3;
 } while (i > 0);
 } } Ans : 10 7 4 1

③ class Test {
 public static void main (String [J] args) {
 int num = 15;
 if (num > 10) {
 if (num % 3 == 0)
 System.out.print("Divisible");
 else
 System.out.print("Not Divisible");
 } } Ans : Divisible

④ class Test {
 public static void main (String [J] args) {
 int a = 7;
 if (a == 7)
 System.out.print("Match");
 else
 System.out.print("No Match"); } } Ans : Match

⑤ public class Main {

 public static void main (String [] args) {

 int i = 1;

 while (i < 10) {

 i += 2;

 }

 System.out.println(i);

 }

Ans: 11

⑥ class Test {

 public static void main (String [] args) {

 int sum = 0;

 for (int i = 1; i <= 5; i++) {

 sum += i;

 }

 System.out.println (sum);

 }

Ans: 15

⑦ class Test {

 public static void main (String [] args) {

 int x = 10;

 if (x > 5) {

 System.out.println ("High");

 }

 }

Ans: High

⑧ class Test {

 public static void main (String [] args) {

 int n = 10;

 for (int i = 1; i <= 20; i = i * 2) {

 System.out.print (i + " ");

 }

}

Ans: 1 2 4 8 16.

WEEK - 2 - SKILL BUILDER

- ① Arun aims to create a project that takes positive integers as marks for five subjects from the user. If the average of the marks is greater than or equal to 50, the student has passed the exam. Otherwise, the student has failed.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main (String [] args) {  
        int n=0, i=5;  
        Scanner scon = new Scanner (System.in);  
        while (i>0) {  
            n += scon.nextInt();  
            i--;  
        }  
        int avg = (int) (n/5.0);  
        System.out.println ("Avg Score: " + avg);  
        if (avg >= 50) {  
            System.out.println ("The student has passed");  
        } else {  
            System.out.println ("The student has failed");  
        }  
    }  
}
```

Input:

50 60 70 80 90

Output: Average score = 70

The student has passed.

- ② Samantha is a diligent math student who is exploring the world of programming. She is learning Java and has recently studied conditional statements. One day, her teacher gives her an interesting problem to solve, which takes a number as input and checks whether it is a multiple of 5 or 7.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main (String [] args) {  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        if ((n % 5 == 0) && (n % 7 != 0)) {  
            System.out.println ("n + \" is a multiple of 5");  
        } else if ((n % 7 == 0) && (n % 5 != 0)) {  
            System.out.println ("n + \" is a multiple of 7");  
        } else {  
            System.out.println ("n + \" is neither multiple of 5 nor 7");  
        }  
    }  
}
```

Input:

21

Output:

21 is a multiple of 7.

Input:

37

Output: 37 is neither multiple of 5 nor 7.

③

Maya, a student in an arts and crafts class, wants to create a pattern using stars (*) in a specific format. She plans to use a program to help her construct the pattern. Write a program that takes an integer as input and constructs the following pattern using nested for loops.

Program:

```
import java.util.Scanner;  
class Main{  
    public static void main(String [] args){  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        for(int i=1; i<=n; i++){  
            for(int j=i; j>0; j--){  
                System.out.print (" * ");  
            }  
            System.out.print ("\n");  
        }  
        for(int i=1; i<=n; i++){  
            for(int j=n-i; j>0; j--){  
                System.out.print (" * ");  
            }  
            System.out.println ();  
        }  
    }  
}
```

Input :

5

Output

```
*  
* *  
* * *  
* * * *  
* * * * *  
* * * * * *  
* * * * * * *  
* * * * * * * *  
* * * * * * * * *  
* * * * * * * * * *
```

④ Write a program that employees to weave this captivally numerical pyramid as shown below.

Input : 4

Output :

```
    1  
   1 2 3  
  1 2 3 4 5  
1 2 3 4 5 6 7
```

Program :

```
import java.util.Scanner;  
class Main {  
    public static void main (String [] args) {  
        Scanner sc = new Scanner (System.in);  
        int c,;  
        int h = sc.nextInt();  
        while (h > 0) {  
            int s = h - 1;  
            while (s > 0) {  
                System.out.print(" ");  
                s--;  
            }  
            for (int i = 1; i <= c; i++) {  
                System.out.print ("%d", i);  
            }  
            System.out.print (" ");  
            h--;  
            c = c + 2;  
        }  
    }  
}
```

- ① The program should print the zigzag pattern with a tab separating the columns. For each row, the input numbers should follow a diagonal pattern where numbers are placed in a zigzag, left to right on odd rows and right to left on even rows.

Input: 3

Output:

2 4

(Bridged) 5 6

Program:

```
import java.util.Scanner;
public class Main {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        int N = sc.nextInt ();
        for (int r=0; r<N; r++) {
            for (int t=0; t<(N-r); t++) {
                System.out.print ("\t");
            }
            for (int c=0; c<r; c++) {
                int base = c*N - (c*(c-1))/2;
                int val = base + (r-c+1);
                System.out.print (val);
            }
            if (c < r) System.out.print ("\t");
        }
        System.out.println ();
    }
}
```

② Write a program that accepts a numeric grade as input. The program should then convert this numeric grade into a letter grade based on specific conditions. The letter grades are A, B, C, D and F.

Program :

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner (System.in);  
        int grade = sc.nextInt();  
        String letter;  
        if (grade >= 90) letter = "A";  
        else if (grade >= 80) letter = "B";  
        else if (grade >= 70) letter = "C";  
        else if (grade >= 60) letter = "D";  
        else letter = "F";  
        System.out.println ("Letter grade : " + letter);  
    }  
}
```

Input :

45

Output :

Letter grade : F

③ Write a program to calculate the total bill using the rules: The program should output the total bill after applying the discount with two decimal places.

program :

```
import java.util.Scanner;  
class Main {  
    public static void main (String [] args) {  
        Scanner sc = new Scanner (System.in);  
        double electricityUnits = sc.nextDouble();  
        double waterUnits = sc.nextDouble();  
        double gasUnits = sc.nextDouble();  
        double totalCost = (electricityUnits * 0.12) +  
                           (waterUnits * 0.05) +  
                           (gasUnits * 0.08);  
        if (totalCost >= 100) {  
            totalCost -= totalCost * 0.10;  
        } else if (totalCost >= 50) {  
            totalCost -= totalCost * 0.05;  
        }  
        System.out.printf ("% .2f", totalCost);  
    }  
}
```

Input :

500.0
30.0
20.0

Output :

59.95,

- (0) Raj is solving a physics problem involving projectile motion, where he needs to calculate the time a ball hits the ground using a quadratic equation of the form $ax^2 + bx + c = 0$. Depending on the coefficients, the ball may hit the ground once, twice or not at all in real time.

Programs

```

import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        int a = sc.nextInt();
        int b = sc.nextInt();
        int c = sc.nextInt();
        int d = (b * b) - (4 * a * c);
        if (d > 0) {
            double root1 = (-b + Math.sqrt(d)) / (2 * a));
            double root2 = (-b - Math.sqrt(d)) / (2 * a));
            System.out.println ("Two real Solutions:");
            System.out.printf ("Root1 = %.2f\n", root1);
            System.out.printf ("Root2 = %.2f", root2);
        } else if (d == 0) {
            double root = -b / (2 * a));
            System.out.println ("One real Solution:");
            System.out.printf ("Root = %.2f", root);
        } else {
            System.out.println ("There are no real
                solution.");
        }
    }
}

```

② Samantha is a diligent math student who is exploring the world of programming. She is learning Java and has recently studied conditional statement interesting problem to solve, which takes a number as input and checks whether it is a multiple of 5 or 7.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
        if ((n % 5 == 0) && (n % 7 != 0)) {  
            System.out.print(" %d is a multiple of 5", n);  
        } else if ((n % 7 == 0) && (n % 5 != 0)) {  
            System.out.printf("%d is a multiple of 7", n);  
        } else {  
            System.out.println("%d is neither multiple of 5 nor 7", n);  
        }  
    }  
}
```

Input

21

Output

21 is a multiple of 7.

WEEK-3-MCA

① class Test {

```
public static void main (String [] args) {
    int [] x = {3, 8, 12};
    int result = x[0] + x[2];
    System.out.println(result);
}
```

3

Ans : 48

② class Sample {

```
public static void main (String [] args) {
    int [][] matrix = {
        {1, 2, 3},
        {4, 5, 6}
    };
    System.out.println(matrix[1][2]);
}
```

3

Ans : 6

③ class A {

```
public static void main (String [] args) {
    int [] a = {1, 2, 3, 4};
    for (int i=0; i < a.length/2; i++) {
        int temp = a[i];
        a[i] = a[a.length - 1 - i];
        a[a.length - 1 - i] = temp;
    }
    System.out.print(a[0]);
}
```

3 }

Ans : 4

F

④ class A {

```
public static void main (String [] args) {
    int [][] a = {{1, 2}, {3, 4}};
    for (int i=0; i < a.length; i++) {
        for (int j=0; j < a[0].length; j++) {
            System.out.print(a[i][j] + " ");
        }
    }
}
```

Ans : 1 2 3

⑥ class Sample {

```
public static void main (String [] args) {
    int [] a = {1, 2, 3};
    int product = 1;
    for (int i=0; i < a.length; i++) {
        product *= a[i];
    }
    System.out.println (product);
}
```

Soln: 6

⑦ class Q {

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

```
    int [] a = {1, 2, 3, 4};
    for (int i=0; i < a.length; i++) {
        if (a[i] % 2 == 0) {
            a[i] = 0;
        }
    }
    System.out.println (a[0] + " " + a[3]);
}
```

Aus: 0 0

⑧ class Q {

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

```
    int [] arr = {5, 6, 7, 8, 9, 10};
```

```
    System.out.println (arr [0][2]);
}
```

Aus: 7

⑨ class Q {

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

```
    int [] num = {4, 2, 9, 5};
```

```
    int max = num [0];
```

```
    for (int i=1; i < num.length; i++) {
```

```
        if (num [i] > max)
```

```
            max = num [i];
    }
    S.O.P (max);
}
```

Aus: 9

Week-3 - Skill Builder:

① Roh is intrigued by numerical patterns. Today he stumbled upon a puzzle while working with arrays. He wants to compute the sum of third largest and second-smallest element of list.

Program:

```
import java.util.*;  
class Main {  
    public static void main (String [] args) {  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        int [] arr = new int [n];  
        for (int i=0; i<n; i++) {  
            arr[i] = scan.nextInt();  
        }  
        Arrays.sort (arr);  
        System.out.println (arr[0] + arr[n-3]);  
    }  
}
```

Input

3

10 28 4 7

Output

38

② Write a code to calculate the sum of the weights of the first and last packages in the list. The program should take an integer array as input and return the total weight.

Program :

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        int n = sc.nextInt();  
        int[] arr = new int[n];  
        for (int i = 0; i < n; i++) {  
            arr[i] = sc.nextInt();  
        }  
        System.out.println("Sum " + (arr[0] + arr[n - 1]));  
    }  
}
```

Input :

5
10 20 30 40 50

Output

Sum : 60

③ Main() is interested in finding a treasure but the key to opening is to get the sum of the main diagonal elements and secondary diagonal elements.

Program:

```
import java.util.Scanner;  
class Main  
{  
    public static void main (String [] args) {  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        int [] [] arr = new int [n] [n];  
        for (int i=0; i<n; i++) {  
            for (int j=0; j<n; j++) {  
                arr [i] [j] = scan.nextInt();  
            }  
        }  
        int md_sum = 0, sd_sum = 0;  
        for (int i=0; i<n; i++) {  
            for (int j=0; j<n; j++) {  
                if (i==j) {  
                    md_sum += arr [i] [j];  
                }  
                if (i+j == n-1) {  
                    sd_sum += arr [i] [j];  
                }  
            }  
        }  
        System.out.println ("MD-SUM : " + md_sum);  
        System.out.println ("SD-SUM : " + sd_sum);  
    }  
}
```

Output :

1 2 3

Output 10

① Egath is participating in a coding hackathon, and one of the challenges requires him to work with an array of integers. The task is to remove exactly one element from array such that the sum of remaining elements is a prime number.

Program:

```
import java.util.*;
```

```
class Main {
```

```
    public static boolean IsPrime (int n) {
```

```
        if (n == 2) {
```

```
            return true;
```

```
        } else if (n < 2) {
```

```
            return false;
```

```
    } else {
```

```
        for (int i = 2; i < n; i++) {
```

```
            if (n % i == 0) {
```

```
                return false;
```

```
}
```

```
        return true;
```

```
}
```

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

```
        Scanner scan = new Scanner (System.in);
```

```
        int n = scan.nextInt();
```

```
        boolean flag = false;
```

```
        int [] arr = new int [n]; int sum = 0;
```

```
        for (int i = 0; i < n; i++) {
```

```
            arr [i] = scan.nextInt();
```

```
            sum += arr [i];
```

```
}
```

```
        for (int i = 0; i < n; i++) {
```

```
            if (IsPrime (sum - arr [i])) {
```

```
                System.out.print (sum - arr [i]);
```

Output

3

1 2 3

Output

5

- ③ Write a program to print the array of reward points, removing any duplicate while preserving the order of unique entries. The program should display clean list of count.

Program:

```
import java.util.*;  
class Main{  
    public static void main (String [] args){  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        int [] arr = new int [n];  
        for (int i=0; i<n; i++){  
            arr[i] = scan.nextInt();  
        }  
        int k=0;  
        int [] res = new int [n];  
        Arrays.fill (res, 0);  
        for (int i=0; i<n; i++){  
            boolean flag = true;  
            for (int j=0; j<n; j++){  
                if (arr[i] == res[j]) {  
                    flag = false;  
                    break;  
                }  
            }  
            if (flag){  
                res[k] = arr[i];  
                k++;  
            }  
        }  
        int i=0;  
        while (res[i] != 0){  
            System.out.print (res[i] + " ");  
            i++;  
        }  
        System.out.print ("\n", +4);  
    }  
}
```

Input : 100 100 200

Output : 100 200

③ Write a program to find the sum of all the numbers written on each ball in the grid

Program :

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        int m = scan.nextInt();  
        int n = scan.nextInt();  
        int s = 0;  
  
        for (int i = 0; i < m; i++) {  
            for (int j = 0; j < n; j++) {  
                s += scan.nextInt();  
            }  
            System.out.print(s);  
        }  
    }  
}
```

Input :

3

3

1 2 3

4 5 6

7 8 9

Output :

45

① Imagine you have an array of integer values and you're tasked with identifying a pair of elements that have a sum that is the closest to zero when compared to any other pair of array.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        int n = scan.nextInt();  
        int[] arr = new int[n];  
        for (int i = 0; i < n; i++) {  
            arr[i] = scan.nextInt();  
        }  
        int s1 = 0, s2 = 0, sum = 500;  
        for (int i = 0; i < n; i++) {  
            for (int j = i + 1; j < n; j++) {  
                if (Math.abs(arr[i] + arr[j]) < sum) {  
                    s1 = arr[i];  
                    s2 = arr[j];  
                    sum = Math.abs(s1 + s2);  
                }  
            }  
        }  
        System.out.println("Pair with the sum  
        closest to zero : " + s1 + " and " + s2);  
    }  
}
```

Input:

4

2 -4 1 6

Output

Pair with the sum closest to zero : -4 and 6

② Your task is to help Purna implement a Java Program that takes this representation as input and rotates it within the same structure.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        int n = scan.nextInt();  
        int[][] arr = new int[n][n];  
        for (int i = 0; i < n; i++) {  
            for (int j = 0; j < n; j++) {  
                arr[i][j] = scan.nextInt();  
            }  
        }  
        System.out.println("Rotated 2D Array");  
        for (int j = 0; j < n; j++) {  
            for (int i = n - 1; i >= 0; i--) {  
                System.out.print(arr[i][j] + " ");  
            }  
            System.out.println();  
        }  
    }  
}
```

Input :

```
3  
1 2 3  
4 5 6  
7 8 9
```

Output :

Rotated 2D Array :

```
4 1  
5 2  
6 3
```

③ Robin is working on a project to find special elements in an array called 'leaders'. Leaders are those exceptional elements that are greater than the sum of elements to its right.

Program:

```
import java.util.Scanner;
class Main {
    public static void main (String [] args) {
        Scanner scan = new Scanner (System.in);
        int n = scan.nextInt();
        int [] arr = new int [n];
        for (int i = 0; i < n; i++) {
            arr [i] = scan.nextInt();
        }
        for (int x = 0; x < n; x++) {
            int sum = 0;
            for (int y = x + 1; y < n; y++) {
                sum += arr [y];
            }
            if (sum < arr [x]) {
                System.out.print (arr [x]);
            }
        }
    }
}
```

Output:

6 28 74 19 25 11

① public class Main {
 public static void main (String [] args) {
 String a = "java";
 char temp = a.charAt(1);
 System.out.println(temp); } } Aus : a

② class Main {
 public static void main (String [] args) {
 String greet = "Welcome \n";
 System.out.print ("String : " + greet);
 int length = greet.length();
 System.out.print ("Length : " + length); } } Aus : String : Welcome
Length : 8

③ class Main {
 public static void main (String [] args) {
 String [] languages = {"C", "C++", "Java", "Python", "Ruby"};
 for (String s : languages) {
 System.out.println (s); } } } Aus : C
C++
Java
Python
Ruby

④ class Main {
 public static void main (String [] args) {
 String s1 = "Hello Java-Program";
 String s2 = new String (s1);
 System.out.print (s1 == s2) + " " + s1.equals (s2); } } Aus : false true

⑤ class Main {
 public static void main (String [] args) {
 StringBuffer c = new StringBuffer ("Hello"); } }

⑥ public class Main {

```
    public static void main (String [] args) {
        float a = 10.0f;
        String temp = Float.toString(a);
        System.out.println (temp);
    }
}
```

Ans: 10.0

⑦ public class Main {

```
    public static void main (String [] args) {
        String str = "1234,34";
        int a = Integer.parseInt(str);
        System.out.println (a);
    }
}
```

Ans: NumberFormat exception

⑧ class Main {

```
    public static void main (String args[]) {
        StringBuffer sb = new StringBuffer ("Java!");
        sb.append ("like ");
        System.out.println (sb);
    }
}
```

Ans: Java! like a!

⑨ class Main {

```
    public static void main (String [] args) {
        String name = "Work Hard";
        name.concat (" Success");
        System.out.println (name);
    }
}
```

Ans: Work Hard Success

Rosh
of H
Prog
imp
cl

is intrigued by numerology. She wants to compute the sum third-largest and second-smallest elements from her.

Program:

```
import java.util.*;  
class Main {  
    public static void main (String [] args) {  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        int [] arr = new int [n];  
        for (int i=0; i<n; i++) { arr[i] = scan.nextInt(); }  
        Arrays.sort (arr);  
        System.out.println (arr[1] + arr[n-3]);  
    }  
}
```

Input:

3
10 28 47

Output:

38

Monica is interested in finding a formula but the key to opening is to get the sum of the main diagonal elements and secondary diagonal elements. Write a program to help Monica find the diagonal sum of a square 2D array.

Program:
 import java.util.Scanner;
 class Main {
 public static void main(String[] args) {
 Scanner scan = new Scanner(System.in);
 int n = scan.nextInt();
 int arr[] = new int[n];
 for (int i = 0; i < n; i++) {
 for (int j = 0; j < n; j++) {
 if (i == j) System.out.print(" ");
 else if (i + j == n - 1) System.out.print("*");
 else System.out.print("#");
 }
 }
 }
 }

2

Output:

1 2 3
 4 5 6
 * 8 9

Output:

15
 19

You are developing a warehouse management system for a shopping website.
To verify that the weight capacity is not exceeded, the
program needs to calculate the sum of the weights of the
first and last packages in the list.

Program:

```
import java.util.*;  
class Main {  
    public static void main (String [] args) {  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        int arr [] = new int [n];  
        for (int i=0; i<n; i++) {  
            arr [i] = scan.nextInt();  
        }  
        System.out.println (arr [0] + arr [n-1]);  
    }  
}
```

Input:

9
100 -200 300 -400 500 -600 700 800 -900

Output: - 800

① The task is to remove exactly one element from the array such that the sum of the remaining elements is a prime.

Program:

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

```
class Main {
```

```
    public static boolean IsPrime(int n) {
```

```
        if (n == 2) return true;
```

```
        else if (n < 2) return false;
```

```
    } else {
```

```
        for (int i = 2; i < n; i++) {
```

```
            if (n % i) == 0)
```

```
                return false;
```

```
        } return true;
```

```
}
```

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

```
        Scanner scan = new Scanner (System.in);
```

```
        int n = scan.nextInt();
```

```
        boolean flag = false;
```

```
        int [] arr = new int [n];
```

```
        int sum = 0;
```

```
        for (int i = 0; i < n; i++) {
```

```
            arr [i] = scan.nextInt();
```

```
            sum += arr [i];
```

```
        } for (int i = 0; i < n; i++) {
```

```
            if (IsPrime (sum - arr [i]))
```

```
                System.out.print ("Sum -
```

```
                flag = true;
```

```
                break;
```

```
        }
```

```
        if (!flag) {
```

Output

Output

⑨ Write a program to process the array of rewed points, removing any duplicates while preserving the order of unique entries. The program should then display unique list.

Program:

```
import java.util.*;  
class Main {  
    public static void main(String [] args) {  
        Scanner scan = new Scanner (System.in);  
        int n = scan.nextInt();  
        int [] arr = new int [n];  
        for (int i=0 ; i<n ; i++) { arr[i] = scan.nextInt(); }  
        int u=0;  
        int [] res = new int [n];  
        Arrays.fill (res, 0);  
        for (int i=0 ; i<u ; i++) {  
            boolean flag = true;  
            for (int j=0 ; j<n ; j++) {  
                if (arr[i] == res[j]) {  
                    flag = false;  
                    break; } } }  
            if (flag) {  
                res[u] = arr[i];  
                System.out.print (res[u] + " "); } }  
        System.out.print ("n"); } };
```

Output:

```
3  
100 100 200
```

Output:

```
100 200  
2
```

- ③ Write a program to find the sum of all the numbers written on each ball in the grid.

Program:

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        Scanner scan = new Scanner(System.in);  
        int m = scan.nextInt();  
        int n = scan.nextInt();  
        int s = 0;  
        for (int i = 0; i < m; i++) {  
            for (int j = 0; j < n; j++) {  
                s += scan.nextInt();  
            }  
        }  
        System.out.print(s);  
    }  
}
```

Input :

```
3  
2  
11 2  
12 3  
13 4
```

Output :

45

① A bookstore wants to analyse the titles of books to determine their longest word in each title. This helps in designing banners and covers. Your task is to write a program.

Program :

```

import java.util.Scanner;

public class Main {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        String sentence = sc.nextLine();
        String [] words = sentence.split (" ");
        String longestWord = "";
        for (String word : words) {
            if (word.length () > longestWord.length ()) {
                longestWord = word;
            }
        }
        System.out.println (longestWord);
    }
}
  
```

Input

The Chronicles of Narnia

Output

Chronicles.

- ① In a university library, librarians need to track the usage of special characters in students notes. The symbols of interest are (!, :, ;).

Program:

```
import java.util.Scanner;  
public class Main {  
    public static void main (String [] args){  
        Scanner sc = new Scanner (System.in);  
        int T = Integer.parseInt (sc.nextLine());  
        for (int t=0; t < T; t++){  
            String passage = sc.nextLine();  
            int conte = 0, countc = 0, counts = 0;  
            for (int i=0; i < passage.length(); i++){  
                char ch = passage.charAt(i)  
                if (ch == '!') conte++;  
                else if (ch == ':') countc++;  
                else if (ch == ';') counts++;  
            }  
            System.out.println (conte + " " + countc + " " + counts);  
        }  
    }  
}
```

Output

Hello ! How are you

Output

1 0 0

① class Ball {

 int size = 11;

}

class Main {

 public static void main(String[] args) {

 Ball b1 = new Ball();

 Ball b2 = new Ball();

 b2.size = 10;

 System.out.println(b1.size);

}

Ans : 11

② class Box {

 int volume (int l, int b, int h) {

 return l * b * h;

}

public class Main {

 public static void main (String[] args) {

 Box b = new Box();

 System.out.println(b.volume(2, 3, 4));

}

Ans : 24

③ class Test {

 private int value;

 Test (int value) {

 this.value = value;

}

 public int getValues() {

 return value;

}

Ans : Compile-time-error

public class Main {

 public static void main (String[] args) {

 Test obj = new Test (10);

 System.out.println (obj.value);

}

④ class Demo {
 void printMessage() {
 System.out.println("Hello from Demo");
 } } Ans: Hello from Demo

public class Main {
 public static void main (String [] args) {
 Demo d = new Demo();
 d.printMessage();
 } } Ans: Hello from Demo

⑤ class A { int y = 30; }
 public class Main {
 public static void main (String [] args) {
 A a1 = new A();
 A a2 = new A();
 a1.y = 50
 System.out.println(a2.y);
 } } Ans: 30

⑥ class Box {
 int length = 5;
 int width = 4;
 int area () {
 return length * width;
 } } Ans: Area = 20
 public static void main (String [] args) {
 Box b = new Box();
 System.out.println("Area = " + b.area());
 } } Ans: Area = 20

⑦ class MathUtils {
 int add (int x) {
 return x+x;
 }
}
3
public class Main {
 public static void main (String [] args) {
 MathUtils m = new MathUtils();
 System.out.println (m.add(3));
 }
}
3

Ans: 10

⑧ class A {
 int val = 20;
}
3
public class Main {
 public static void main (String [] args) {
 A obj1 = new A();
 A obj2 = obj1;
 obj2.val += 5;
 System.out.println (obj1.val);
 }
}
3

Ans: 25

⑨ class Person {
 int age = 18;
}
3
public class Main {
 public static void main (String [] args) {
 Person p = new Person();
 p.age += 2;
 System.out.println ("Age : " + p.age);
 }
}

Ans : Age : 20

- ① You are working as a developer for City Cabs, a taxi service. Customer has Booking ID, Name, distance. Calculate fare and display details.

Program

```

import java.util.Scanner;
class Booking {
    private int bookingId;
    private String customerName;
    private double distance;
    private double fare;
    public Booking (int bookingId, String customerName, double distance) {
        this.bookingId = bookingId;
        this.customerName = customerName;
        this.distance = distance;
    calculateFare();
    }
    public void setBookingID (int bookingId) {
        this.bookingId = bookingId;
    }
    public void setCustomerName (String customerName) {
        this.customerName = customerName;
    }
    public void setDistance (double distance) {
        this.distance = distance;
    calculateFare();
    }
    public int getBookingID () {
        return bookingId;
    }
    public String getCustomerName () {
        return customerName;
    }
    public double getDistance () {
        return distance;
    }
}

```

```

public double getFare() {
    return fare;
}

private void calculateFare() {
    fare = 50 + distance * 10;
    if (distance > 50) {
        fare = fare - (fare * 0.1);
    }
}

class CityCabApp {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        for (int i = 0; i < n; i++) {
            int id = sc.nextInt();
            String name = sc.nextLine();
            double distance = Double.parseDouble(sc.nextLine());
            Booking booking = new Booking(id, name, distance);
            System.out.println("Booking ID: " + booking.getId());
            System.out.println("Customer Name: " + booking.getCustomerName());
            System.out.println("Final Fare: " + booking.getFare());
        }
    }
}

```

Input

1
1234

Rahul Sharma

15

Output:

Booking ID: 1234

Customer Name: Rahul Sharma

Final fare: 200.0

- Q) Nisha is working as a developer for City Electricity Board, which wants to build a household electricity billing system. Generate bill and display each customer's details and final bill amount.

Program :

```
import java.util.Scanner;  
class Customer {  
    private int customerID;  
    private String customerName;  
    private double unitsConsumed;  
    public Customer (int customerID, String customerName, double unitsConsumed) {  
        this.customerID = customerID;  
        this.customerName = customerName;  
        this.unitsConsumed = unitsConsumed;  
    }  
    public void setCustomerID (int customerID) {  
        this.customerID = customerID;  
    }  
    public void setCustomerName (String customerName) {  
        this.customerName = customerName;  
    }  
    public void setUnitsConsumed (double unitsConsumed) {  
        this.unitsConsumed = unitsConsumed;  
    }  
    public int getCustomerID () {  
        return customerID;  
    }  
    public String getCustomerName () {  
        return customerName;  
    }  
    public double getUnitsConsumed () {  
        return unitsConsumed;  
    }
```

```

public double calculateBill() {
    double bill;
    if (CustomerCount <= 100) {
        bill = Unstamped * 5;
    } else if (CustomerCount <= 200) {
        bill = (100 + 5) + ((CustomerCount - 100) * 7);
    } else {
        bill = (100 + 5) + (100 * 7) + ((CustomerCount - 200) * 10);
    }
}

```

}

class Main {

```

public static void main (String [] args) {
    Scanner sc = new Scanner (System.in);
    int n = sc.nextInt();
    for (int i = 0; i < n; i++) {
        int id = sc.nextInt();
        String name = sc.nextLine();
        double unstamp = sc.nextDouble();
        Customer c = new Customer (id, name, unstamp);
        System.out.print (c.calculateBill());
    }
}

```

Input :

100 2

Sita Sharma

103

Output

521.0

(1) Ajali is working as a developer for City Fitness Gym. Build a system to calculate monthly membership fees for gym members. Display each member's details.

Program:

```

import java.util.*; + (2 * 500) = 1100
class Member { } ads
    int m_id; + (2 * 500) = 1100
    String m_name; + (2 * 500) = 1100
    String m_type; + (2 * 500) = 1100
    int services; + (2 * 500) = 1100
    public Member (int m_id, String m_name, String m_type, int services) + (2 * 500) = 1100
        this.m_id = m_id; + (2 * 500) = 1100
        this.m_name = m_name; + (2 * 500) = 1100
        this.m_type = m_type; + (2 * 500) = 1100
        this.services = services; + (2 * 500) = 1100
    }
    public void setID (int m_id){ + (2 * 500) = 1100
        this.m_id = m_id; + (2 * 500) = 1100
    }
    public void setName (String m_name){ + (2 * 500) = 1100
        this.m_name = m_name; + (2 * 500) = 1100
    }
    public void setType (String m_type){ + (2 * 500) = 1100
        this.m_type = m_type; + (2 * 500) = 1100
    }
    public void setServices (int services){ + (2 * 500) = 1100
        this.services = services; + (2 * 500) = 1100
    }
    public double calcFee (String m_type, int services){ + (2 * 500) = 1100
        double fee = 5000; + (2 * 500) = 1100
        switch (m_type){ + (2 * 500) = 1100
            case "Basic": + (2 * 500) = 1100
                fee += 1000.0; + (2 * 500) = 1100
                break; + (2 * 500) = 1100
            case "Premium": + (2 * 500) = 1100
                fee += 1500.0; + (2 * 500) = 1100
                break; + (2 * 500) = 1100
        }
        return fee; + (2 * 500) = 1100
    }
}

```

```

    pictures and card "Elite":
    fee += 2000.0;
}

if (sessions > 5) {
    fee = fee - (fee * 0.1);
}

if (fee > 4000 && m_type.equals("Elite")) {
    fee = fee + (fee * 0.05);
}

return fee;
}
}

```

Java Main:

```

public static void main (String [] args) {
    Scanner sc = new Scanner (System.in);
    int n = sc.nextInt();
    for (int i = 0; i < n; i++) {
        int m_id = sc.nextInt();
        String m_name = sc.nextLine();
        String m_type = sc.nextLine();
        System.out.println(m_calFee (m_type, m.services));
    }
}

```

Output

```

1001
Roni
Basic
3

```

Output

2500.0

WEEK - 6

- (①) Elsa subscribes to a premium service with a base monthly cost, a service tax with an extra feature cost. Assist her in writing an inheritance program.

```
class Subscription {
```

```
    double monthlyCost;
```

```
    double serviceTax;
```

```
    double extraFeatureCost;
```

```
}
```

```
class PremiumSubscription extends Subscription {
```

```
PremiumSubscription (double mC, double sT, double eFC) {
```

```
    monthlyCost = mC;
```

```
    serviceTax = sT;
```

```
}                    extraFeatureCost = eFC;
```

```
public double calculateMonthlyCost() {
```

```
}                    return (mC + sT + eFC);
```

```
}
```

Input

10.0

2.5

5.0

Output

Rs , 17 . 50

- (②) Alice is managing an online store and wants to implement a program using inheritance to calculate the selling price.

```
class Product {
```

```
    double price;
```

```
}                    class DiscountProduct extends Product {
```

```
    private double discountRate;
```

```
    DiscountProduct (double p, double drate) {
```

```
        price = p;
```

```
        discountRate = drate;
```

```
}                    public double sellingPrice {
```

Input

50.0

0.20

Output

Rs. 40.00

③ Sales Tax Collector:Properties:

class SalesTaxCalculator {

static double doublePrice;

static double doubleTaxRate;

static int intTaxRate;

static int intPrice;

public static int calculatePrice(int price, int taxRate) {

intPrice = price;

intTaxRate = taxRate;

return (intPrice + ((intPrice * intTaxRate) / 100));

}

public static double calculatePrice(double p, double tr) {

double Price = p;

double TaxRate = tr;

return (doublePrice + ((doublePrice * doubleTaxRate) / 100));

}

Input

100

10

100.0

5.0

Output

110

105.00

(④) Mr. Kapoor wants to create a program to calculate the volume
of cube and cuboid

class Cuboid {

 double length, width, height;

 Cuboid (double length, double width, double height) {

 this.length = length;

 this.width = width;

 this.height = height;

 }

 public double calculateVolume() {

 return (length * width * height);

}

}

 double cubeSide;

 Cube (double cubeSide) {

 super(cubeSide, cubeSide, cubeSide);

}

}

Input

60.0 60.0 60.0

50.0

Output

Volume of Cuboid : 216000.00

Volume of Cube : 125000.00

Week - 7 :

- ① Taheer is working on a health monitoring using an interface called HealthCalculator. It should have a method called CalculateBMI.

Program:

Interface HealthCalculator {

```
    double calculateBMI (double weight, double height);
```

}

class BMIcalculator implements HealthCalculator {

```
    public double calculateBMI (double weight, double height) {
        if (weight == 0 || height == 0)
            return -1;
        return weight / (height * height);
    }
}
```

Input:

70.0

1.75

Output:

BMI : 22.86

②

Interest = Principal × Rate × Time / 100

Implement this using InterestCalculator interface and then SimpleInterestCalculator.

Interface InterestCalculator {

```
    double simpleInterest (double principal, double rate, int time);
```

}

class SimpleInterestCalculator implements InterestCalculator {

```
    public double simpleInterest (double principal, double rate, int time) {
        return (principal * rate * time) / 100;
    }
}
```