Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_2028_OOPS using Java_Week 3_MCQ

Attempt : 1
Total Mark : 15
Marks Obtained : 14

Section 1 : MCQ

1. What will be the output of the following code?

```
public class Test {
    public static void main(String[] args) {
        int[] x = {4, 8, 12};
        int result = x[0] * x[2];
        System.out.println(result);
    }
}
```

Answer

48

Status : Correct *Marks : 1/1*

2. What will be the output of the following code?

```
class Q {
   public static void main(String[] args) {
      int[] nums = {3, 6, 7, 2, 8};
      int sum = 0;
      for (int i = 0; i < nums.length; i++) {
         if (nums[i] % 2 == 0)
            sum += nums[i];
      }
      System.out.println(sum);
   }
}
```

Answer

16

Status : Correct ***Marks : 1/1***

3. What will be the output of the following code?

```
class Q {
   public static void main(String[] args) {
      int[] nums = {4, 2, 9, 5};
      int max = nums[0];
      for (int i = 1; i < nums.length; i++) {
         if (nums[i] > max)
            max = nums[i];
      }
      System.out.println(max);
   }
}
```

Answer

9

Status : Correct ***Marks : 1/1***

4. What will be the output of the given code?

```
public class Main {
   public static void main(String[] args) {
      int[] arr = {1, 2, 3, 4, 5};
      int n = arr.length;
      int temp = arr[0];

      for (int i = 0; i < n - 1; i++) {
         arr[i] = arr[i + 1];
      }
      arr[n - 1] = temp;

      for (int num : arr) {
         System.out.print(num + " ");
      }
   }
}
```

Answer

2 3 4 5 1

Status : Correct ***Marks : 1/1***

5. What will be the output of the following code?

```
class Q {
   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.println(a[0]);
   }
}
```

Answer

1

Status : Wrong *Marks : 0/1*

6. What will be the output of the following code?

```
class M {
    public static void main(String[] args) {
        int[][] arr = {
            {1, 2},
            {3, 4},
            {5, 6}
        };

        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i][0] + " ");
        }
    }
}
```

Answer

1 3 5

Status : Correct *Marks : 1/1*

7. What will be the output of the following code?

```
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);
    }
}
```

Answer

6

Status : Correct *Marks : 1/1*

8. What will be the output of the following code?

```
class ReverseArray {
   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;
      }
      for (int i : a)
         System.out.print(i + " ");
   }
}
```

Answer

4 3 2 1

Status : Correct *Marks : 1/1*

9. What will be the output of the following code?

```
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[1] + " " + a[3]);
   }
}
```

Answer

0 0

Status : Correct *Marks : 1/1*

10. What will be the output of the following code?

```
class Sample {
   public static void main(String[] args) {
      int[][] data = {
         {1, 2},
         {3, 4}
      };
      int sum = 0;

      for (int[] row : data) {
         for (int val : row) {
            sum += val;
         }
      }

      System.out.println("Sum = " + sum);
   }
}
```

Answer

Sum = 10

Status : Correct ***Marks : 1/1***

11. What will be the output of the following code?

```
class Q {
   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] + " ");
         }
      }
   }
}
```

Answer

1 2 3 4

Status : Correct ***Marks : 1/1***

12. What will be the output of the following code?

```
class Q {
   public static void main(String[] args) {
      int[][] arr = {
         {5, 6, 7},
         {8, 9, 10}
      };
      System.out.println(arr[0][2]);
   }
}
```

Answer

7

Status : Correct ***Marks : 1/1***

13. What will be the output of the following code?

```
class Q {
   public static void main(String[] args) {
      int[] a = {1, 2, 1, 3, 1, 4};
      int count = 0;
      for (int i = 0; i < a.length; i++) {
         if (a[i] == 1) count++;
      }
      System.out.println(count);
   }
}
```

Answer

3

Status : Correct *Marks : 1/1*

14. What will be the output of the following code?

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

Answer

10

Status : Correct *Marks : 1/1*

15. What will be the output of the following code?

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

Answer

6

Status : Correct *Marks : 1/1*

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 3_Q1

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Rosh is intrigued by numerical patterns. Today, she stumbled upon a puzzle while working with arrays. She wants to compute the sum of the third-largest and second-smallest elements from a list of integers. She seeks your help to implement a program that solves this for her efficiently.

Input Format

The first line of input is an integer N, representing the size of the array.

The second line of input consists of N space-separated integers, representing the elements of the array.

Output Format

The output displays a single integer representing the sum of the third-largest and second-smallest elements in the array.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10
10 20 30 40 50 60 70 80 90 100

Output: 100

Answer

```
import java.util.*;

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

      // Read size of array
      int N = sc.nextInt();
      int[] arr = new int[N];

      // Read elements
      for (int i = 0; i < N; i++) {
         arr[i] = sc.nextInt();
      }

      // Sort array
      Arrays.sort(arr);

      // Find second smallest and third largest
      int secondSmallest = arr[1];
      int thirdLargest = arr[N - 3];

      // Compute sum
      int result = secondSmallest + thirdLargest;

      // Print result
      System.out.println(result);
   }
}
```

Status : Correct *Marks : 10/10*

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 3_Q2

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Monica 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.

Write a program to help Monica find the diagonal sum of a square 2D array.

Note: The main diagonal of the array consists of the elements traversing from the top-left corner to the bottom-right corner. The secondary diagonal includes elements from the top-right corner to the bottom-left corner.

Input Format

The first line of input consists of an integer N, representing the number of rows and columns.

The following N lines consist of N space-separated integers, representing the 2D array elements.

Output Format

The first line of output prints "Sum of the main diagonal: " followed by an integer, representing the sum of the main diagonal.

The second line prints "Sum of the secondary diagonal: " followed by an integer, representing the sum of the secondary diagonal.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3
1 2 3
4 5 6
7 8 9

Output: Sum of the main diagonal: 15
Sum of the secondary diagonal: 15

Answer

```
// You are using Java
import java.util.*;

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

      // Read size of the matrix
      int N = sc.nextInt();
      int[][] matrix = new int[N][N];

      // Read matrix elements
      for (int i = 0; i < N; i++) {
         for (int j = 0; j < N; j++) {
            matrix[i][j] = sc.nextInt();
         }
      }
```

```
    int mainDiagonalSum = 0;
    int secondaryDiagonalSum = 0;

    // Calculate sums
    for (int i = 0; i < N; i++) {
      mainDiagonalSum += matrix[i][i];
      secondaryDiagonalSum += matrix[i][N - 1 - i];
    }

    // Print results
    System.out.println("Sum of the main diagonal: " + mainDiagonalSum);
    System.out.println("Sum of the secondary diagonal: " +
secondaryDiagonalSum);
  }
}
```

Status : Correct ***Marks : 10/10***

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 3_Q3

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

You are developing a warehouse management system for a shipping company. The system uses an integer array to represent the weights of packages in a specific order. 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.

Task:

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 of the first and last packages.

Input Format

The first line of the input is an integer N representing the size of the array.

The second line of the input is N space-separated integer values.

Output Format

The output is displayed in the following format:

"Sum of the first and last elements: <<Sum>>"

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
10 20 30 40 50
Output: Sum of the first and last elements: 60

Answer

```
import java.util.*;

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

        // Read size of array
        int N = sc.nextInt();
        int[] arr = new int[N];

        // Read array elements
        for (int i = 0; i < N; i++) {
            arr[i] = sc.nextInt();
        }

        // Sum of first and last element
        int sum = arr[0] + arr[N - 1];

        // Print result
        System.out.println("Sum of the first and last elements: " + sum);
    }
}
```

Status : Correct *Marks : 10/10*

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 3_Q4

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Sesha is developing a weather monitoring system for a region with multiple weather stations. Each weather station collects temperature data hourly and stores it in a 2D array.

Write a program that can add the temperature data from two different weather stations to create a combined temperature record for the region.

Input Format

The first line of input consists of two space-separated integers N and M, representing the number of rows and columns of the matrices, respectively.

The next N lines consist of M space-separated integers, representing the values of the first matrix.

The following N lines consist of M space-separated integers, representing the values of the second matrix.

Output Format

The output prints the addition of the two matrices in N rows and M columns, representing the combined temperature record.

Refer to the sample output for formatting specifications.

Sample Test Case

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

```
Output: 2 3 4
6 7 8
10 11 12
```

Answer

```
import java.util.*;

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

        // Read dimensions
        int N = sc.nextInt();
        int M = sc.nextInt();

        int[][] mat1 = new int[N][M];
        int[][] mat2 = new int[N][M];
        int[][] result = new int[N][M];

        // Read first matrix
        for (int i = 0; i < N; i++) {
            for (int j = 0; j < M; j++) {
                mat1[i][j] = sc.nextInt();
            }
        }

        // Read second matrix
        for (int i = 0; i < N; i++) {
            for (int j = 0; j < M; j++) {
                mat2[i][j] = sc.nextInt();
            }
        }

        // Perform addition
        for (int i = 0; i < N; i++) {
            for (int j = 0; j < M; j++) {
                result[i][j] = mat1[i][j] + mat2[i][j];
            }
        }

        // Print result
        for (int i = 0; i < N; i++) {
            for (int j = 0; j < M; j++) {
                System.out.print(result[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Status : Correct ***Marks : 10/10***

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 3_Q5

Attempt : 1
Total Mark : 10
Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Sharon is creating a program that finds the first repeated element in an integer array. The program should efficiently identify the first element that appears more than once in the given array. If no such element is found, it should appropriately display a message.

Help Sharon to complete the program.

Input Format

The first line of input consists of an integer n, representing the number of elements in the array.

The second line consists of n space-separated integers, representing the array elements.

Output Format

If a repeated element is found, print the first element that appears more than once.

If no repeated element is found, print "No repeated element found in the array".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 8
12 21 13 14 21 36 47 21

Output: 21

Answer

```
import java.util.*;

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

        // Read size of array
        int n = sc.nextInt();
        int[] arr = new int[n];

        // Read array elements
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }

        // Use HashSet to track seen elements
        Set<Integer> seen = new HashSet<>();
        boolean found = false;

        for (int num : arr) {
            if (seen.contains(num)) {
                System.out.println(num);
                found = true;
                break;  // stop at the first repeated element
            }
            seen.add(num);
        }

        if (!found) {
            System.out.println("No repeated element found in the array");
        }
    }
}
```

Status : Correct ***Marks : 10/10***

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_2028_OOPS using Java_Week 3_PAH

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Priya is building a system to automate image transformations using matrix operations. To do this, she needs to multiply two matrices representing pixel data and transformation rules.

Help Priya perform matrix multiplication and print the resulting matrix if the operation is valid.

Input Format

The first line of input consists of two int values, representing the number of rows R1 and columns C1 of the first matrix.

The next R1 × C1 integers represent the elements of the first matrix.

The next line consists of two int values, representing the number of rows R2 and columns C2 of the second matrix.

The next R2 × C2 integers represent the elements of the second matrix.

Output Format

If matrix multiplication is possible, print R1 lines, each containing C2 space-separated int values representing the resulting matrix.

Otherwise, print "Matrix multiplication not possible".

Refer to the sample output for formatting specifications.

Sample Test Case

```
Input: 2 3
1 2 3
4 5 6
3 2
7 8
9 10
11 12
Output: 58 64
139 154
```

Answer

```
import java.util.*;

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

      // Read first matrix dimensions
      int R1 = sc.nextInt();
      int C1 = sc.nextInt();
      int[][] A = new int[R1][C1];

      // Read first matrix
      for (int i = 0; i < R1; i++) {
         for (int j = 0; j < C1; j++) {
            A[i][j] = sc.nextInt();
         }
      }

      // Read second matrix dimensions
      int R2 = sc.nextInt();
      int C2 = sc.nextInt();
      int[][] B = new int[R2][C2];

      // Read second matrix
      for (int i = 0; i < R2; i++) {
         for (int j = 0; j < C2; j++) {
            B[i][j] = sc.nextInt();
         }
      }

      // Check multiplication possibility
      if (C1 != R2) {
         System.out.println("Matrix multiplication not possible");
         return;
      }

      // Result matrix
      int[][] result = new int[R1][C2];

      // Matrix multiplication
      for (int i = 0; i < R1; i++) {
         for (int j = 0; j < C2; j++) {
            for (int k = 0; k < C1; k++) {
                result[i][j] += A[i][k] * B[k][j];
            }
         }
      }

      // Print result
      for (int i = 0; i < R1; i++) {
         for (int j = 0; j < C2; j++) {
            System.out.print(result[i][j] + " ");
         }
         System.out.println();
      }
   }
}
```

Status : Correct *Marks : 10/10*

2. Problem Statement

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 the array such that the sum of the remaining elements is a prime number.

Help Egath find the first possible prime sum by removing one element or determining if no such prime sum can be achieved.

Input Format

The first line of input consists of an integer N, representing the number of elements in the array.

The second line consists of N space-separated integers, representing the array elements.

Output Format

If removing one element results in a prime sum, print the sum.

If no such prime sum can be achieved by removing exactly one element, print "No valid prime sum found".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3
1 2 3

Output: 5

Answer

```
// You are using Java
import java.util.*;
```

```
public class Main {

    // Function to check if a number is prime
    static boolean isPrime(int num) {
        if (num <= 1) return false;
        if (num == 2) return true;
        if (num % 2 == 0) return false;
        for (int i = 3; i * i <= num; i += 2) {
            if (num % i == 0) return false;
        }
        return true;
    }

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

        int N = sc.nextInt();
        int[] arr = new int[N];

        int totalSum = 0;
        for (int i = 0; i < N; i++) {
            arr[i] = sc.nextInt();
            totalSum += arr[i];
        }

        boolean found = false;
        for (int i = 0; i < N; i++) {
            int newSum = totalSum - arr[i];
            if (isPrime(newSum)) {
                System.out.println(newSum);
                found = true;
                break; // first possible prime sum
            }
        }

        if (!found) {
            System.out.println("No valid prime sum found");
        }
    }
}
```

Status : Correct ***Marks : 10/10***

3. Problem Statement

Eminem is a billiard player who enjoys playing billiards and also likes solving mathematical puzzles. He notices that the billiard balls on the table are arranged in a grid, and he is curious to find the sum of the numbers written on each ball.

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

Input Format

The first line of input consists of an integer N, representing the number of rows.

The second line consists of an integer M, representing the number of columns.

The following lines N lines consist of M space-separated integers, representing the numbers written on each ball.

Output Format

The output prints an integer representing the sum of all the numbers written on each ball.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 3
3
1 2 3
4 5 6
7 8 9

Output: 45

Answer

```
// You are using Java
import java.util.*;
```

```
public class Main {
  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    // Read dimensions
    int N = sc.nextInt(); // rows
    int M = sc.nextInt(); // columns

    int sum = 0;

    // Read grid and calculate sum
    for (int i = 0; i < N; i++) {
      for (int j = 0; j < M; j++) {
        sum += sc.nextInt();
      }
    }

    // Print result
    System.out.println(sum);
  }
}
```

Status : Correct ***Marks : 10/10***

4. Problem Statement

In a customer loyalty program, reward points are logged in a sorted array as customers make transactions. Occasionally, due to system errors, duplicate entries for the same transaction may appear. To ensure accurate reward calculations, it's crucial to remove these duplicates from the list.

Write a program to process the array of reward points, removing any duplicates while preserving the order of unique entries. The program should then display the cleaned list of unique reward points and the total count of these unique points.

Input Format

The first line of input consists of an integer N, representing the number of reward points.

The second line consists of N space-separated integers, representing the reward points in sorted order.

Output Format

The first line of output prints the cleaned list of unique reward points separated by a space.

The second line of output prints an integer representing the total count of unique reward points.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 3
100 100 200

Output: 100 200
2

Answer

```
import java.util.*;

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

        // Read number of elements
        int N = sc.nextInt();
        int[] arr = new int[N];

        for (int i = 0; i < N; i++) {
            arr[i] = sc.nextInt();
        }

        // Remove duplicates while preserving order
        List<Integer> unique = new ArrayList<>();
        unique.add(arr[0]); // first element always unique

        for (int i = 1; i < N; i++) {
            if (arr[i] != arr[i - 1]) {
                unique.add(arr[i]);
            }
        }

        // Print cleaned list
        for (int num : unique) {
            System.out.print(num + " ");
        }
        System.out.println();

        // Print total count
        System.out.println(unique.size());
    }
}
```

Status : Correct ***Marks : 10/10***

Rajalakshmi Engineering College

Name: Sachin Krishna B
Email: 241901507@rajalakshmi.edu.in
Roll no: 241901507
Phone: 9025753177
Branch: REC
Department: CSE (CS) - Section 2
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_2028_OOPS using Java_Week 3_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Robin is a tech-savvy teenager who is diving into programming.

He 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 all the elements to their right.

Assist Robin in writing this program.

Example

Input:

6

16 28 74 19 25 11

Output:

74 25 11

Explanation:

The element 16 is not greater than the sum of elements to its right (28 + 74 + 19 + 25 + 11 = 157)

The element 28 is not greater than the sum of elements to its right (74 + 19 + 25 + 11 = 129)

The element 74 is greater than the sum of elements to its right (19 + 25 + 11 = 55)

The element 19 is not greater than the sum of elements to its right (25 + 11 = 36)

The element 25 is greater than the sum of elements to its right (11)

The last element 11 is always a leader since there are no elements to its right.

So, the output is {74, 25, 11}.

Input Format

The first line of input consists of an integer N, representing the number of elements in the array.

The second line consists of N space-separated integers, representing the elements of the array.

Output Format

The output prints the special elements in the given array, that are greater than the sum of all the elements to their right.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
3 4 2 5 1

Output: 5 1

Answer

```
import java.util.*;

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

        // Read input size
        int N = sc.nextInt();
        int[] arr = new int[N];

        for (int i = 0; i < N; i++) {
            arr[i] = sc.nextInt();
        }

        List<Integer> leaders = new ArrayList<>();

        // Traverse the array
        for (int i = 0; i < N; i++) {
            int rightSum = 0;
            for (int j = i + 1; j < N; j++) {
                rightSum += arr[j];
            }

            if (arr[i] > rightSum) {
                leaders.add(arr[i]);
            }
        }

        // Print leaders
        for (int num : leaders) {
            System.out.print(num + " ");
        }
    }
}
```

Status : Correct ***Marks : 10/10***

2. Problem Statement:

Emma, a budding computer vision enthusiast, is working on a challenging image processing project. She has a square image represented as a 2D matrix of integers. As part of a special filter operation, she needs to rotate the image by 90 degrees clockwise, but there's a twist — she must perform the rotation in-place, using no extra space.

This means Emma has to rotate the matrix without creating a new one. Your task is to help her implement a Java program that takes this square matrix as input and rotates it within the same structure.

Can you help Emma efficiently rotate the image so that her project can move to the next stage?

Input Format

The first line of input contains a single integer n, representing the number of rows and columns of the square matrix (i.e., the matrix is of size n x n).

The next n lines each contain n space-separated integers, representing the elements of each row of the 2D array.

Output Format

The first line of output prints "Rotated 2D Array:"

The next n lines of output print the rotated matrix.

Each line contains n space-separated integers representing a row of the rotated matrix.

Refer to the sample output for format specification.

Sample Test Case

Input: 3
1 2 3
4 5 6
7 8 9
Output: Rotated 2D Array:
7 4 1

```
8 5 2
9 6 3
```

Answer

```
// You are using Java
import java.util.*;

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

        // Read size
        int n = sc.nextInt();
        int[][] mat = new int[n][n];

        // Read matrix
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                mat[i][j] = sc.nextInt();
            }
        }

        // Step 1: Transpose the matrix
        for (int i = 0; i < n; i++) {
            for (int j = i + 1; j < n; j++) {
                int temp = mat[i][j];
                mat[i][j] = mat[j][i];
                mat[j][i] = temp;
            }
        }

        // Step 2: Reverse each row
        for (int i = 0; i < n; i++) {
            int left = 0, right = n - 1;
            while (left < right) {
                int temp = mat[i][left];
                mat[i][left] = mat[i][right];
                mat[i][right] = temp;
                left++;
                right--;
            }
        }
        // Output
        System.out.println("Rotated 2D Array:");
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                System.out.print(mat[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Status : Correct ***Marks : 10/10***

3. Problem Statement:

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

Your goal is to create a program that solves this problem efficiently. The program should accept an array of integers and return the pair of elements whose sum is closest to zero.

Input Format

The first line of the input is an integer N representing the size of the array.

The second line of the input contains N space-separated integer values.

Output Format

The output is displayed in the following format:

"Pair with the sum closest to zero: {value} and {value}"

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5
9 10 -3 -5 -2
Output: Pair with the sum closest to zero: 9 and -5

Answer

```
// You are using Java
import java.util.*;

public 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();
        }

        int minSum = Integer.MAX_VALUE;
        int val1 = 0, val2 = 0;

        for (int i = 0; i < N - 1; i++) {
            for (int j = i + 1; j < N; j++) {
                int sum = arr[i] + arr[j];
                if (Math.abs(sum) < Math.abs(minSum)) {
                    minSum = sum;
                    val1 = arr[i];
                    val2 = arr[j];
                }
            }
        }

        System.out.println("Pair with the sum closest to zero: " + val1 + " and " +
val2);
    }
}
```

Status : Correct ***Marks : 10/10***

4. Problem Statement

Emma is a data analyst working with a grid-based system where each cell contains important numerical data. The grid represents spatial data, inventory records, or structured reports that require periodic updates.

Due to system updates and new requirements, Emma needs to modify the grid in the following ways:

She wants to insert either a new row or a new column at a given position.Later, she needs to delete either a row or a column from the modified matrix.

Input Format

The first line contains two integers rows and cols (the dimensions of the matrix).

The next rows lines contain cols space-separated integers representing the initial matrix.

The next line contains two integers insertType and insertIndex:

- insertType = 0 for row insertion, 1 for column insertion.
- insertIndex is the position where the new row/column should be added.

If inserting a row, the next cols integers represent the new row or If inserting a column, the next rows integers represent the new column.

The next line contains two integers deleteType and deleteIndex:

- deleteType = 0 for row deletion, 1 for column deletion.
- deleteIndex is the position to be deleted.

Output Format

The first line of output prints the string "After insertion" followed by the modified matrix with the inserted row or column.

Each row of the matrix is printed on a new line with space-separated integers.

The next line prints the string "After deletion" followed by the final matrix after the specified deletion operation.

Each row of the resulting matrix is printed on a new line with space-separated integers.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3 3
1 2 3
4 5 6
7 8 9
0 1
10 11 12
1 2
Output: After insertion
1 2 3
10 11 12
4 5 6
7 8 9
After deletion
1 2
10 11
4 5
7 8

Answer

```
import java.util.*;

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

        int rows = sc.nextInt();
        int cols = sc.nextInt();

        // Read initial matrix
        List<List<Integer>> matrix = new ArrayList<>();
        for (int i = 0; i < rows; i++) {
            List<Integer> row = new ArrayList<>();
            for (int j = 0; j < cols; j++) {
                row.add(sc.nextInt());
            }
            matrix.add(row);
        }

        // Insertion
        int insertType = sc.nextInt();
        int insertIndex = sc.nextInt();

        if (insertType == 0) { // Row insertion
            List<Integer> newRow = new ArrayList<>();
            for (int j = 0; j < cols; j++) {
                newRow.add(sc.nextInt());
            }
            matrix.add(insertIndex, newRow);
            rows++;
        } else { // Column insertion
            List<Integer> newCol = new ArrayList<>();
            for (int i = 0; i < rows; i++) {
                newCol.add(sc.nextInt());
            }
            for (int i = 0; i < rows; i++) {
                matrix.get(i).add(insertIndex, newCol.get(i));
            }
            cols++;
        }

        System.out.println("After insertion");
        printMatrix(matrix);

        // Deletion
        int deleteType = sc.nextInt();
        int deleteIndex = sc.nextInt();

        if (deleteType == 0) { // Row deletion
            matrix.remove(deleteIndex);
            rows--;
        } else { // Column deletion
            for (int i = 0; i < rows; i++) {
                matrix.get(i).remove(deleteIndex);
            }
            cols--;
        }

        System.out.println("After deletion");
        printMatrix(matrix);

        sc.close();
    }

    private static void printMatrix(List<List<Integer>> matrix) {
        for (List<Integer> row : matrix) {
            for (int val : row) {
                System.out.print(val + " ");
            }
            System.out.println();
        }
    }
}
```

Status : Correct ***Marks : 10/10***