IRC_JAVA_COD_CS_PATTERN

Test Summary

- No. of Sections: 1
- No. of Questions: 10
- Total Duration: 180 min

Section 1 - CODING

Section Summary

- No. of Questions: 10
- Duration: 180 min

Additional Instructions:

None

Q1. **Problem statement:**

write a java program to print this pattern.

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

Input Format

The first input consists of the N value.

Output Format

Refer to the sample output for the pattern to be printed.

Sample Input

```
4
```

Sample Output

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

Sample Input

```
5
```

Sample Output

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

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q2. **Problem statement:**

Write a java program to print this pattern.

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

Input Format

An integer input in first line

Output Format

Refer the sample output

Sample Input

```
5
```

Sample Output

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

Sample Input

```
4
```

Sample Output

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

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q3. **Problem statement:**

Write a java program to print the pattern.

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

Input Format

The input consists of n value.

Output Format

Refer to the sample output for the specifications.

Sample Input

```
5
```

Sample Output

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

Sample Input

```
4
```

Sample Output

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

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q4. **Problem statement:**

Write a java program to print a half pyramid using numbers.

1
1 2
1 2 3
1 2 3 4
1 2 3 4 5

Input Format

The first input consists of N value.

Output Format

The output prints the pattern

Sample Input

```
5
```

Sample Output

```
1
1 2
1 2 3
1 2 3 4
```

Sample Input

```
4
```

Sample Output

```
1
1 2
1 2 3
1 2 3 4
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q5. **Problem statement:**

Write a program to print half pyramid using alphabets.

Input: 5

Output
A
B B
C C C
D D D D
E E E E E

Input Format

Number of lines

Output Format

Refer to the sample output for the specifications.

Sample Input

```
5
```

Sample Output

```
A
B  B
C  C  C
D  D  D  D
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q6. **Problem statement:**
Write a java program to print the pascal's triangle.

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

Input Format

The first input integer is the number of rows

Output Format

The output prints the pascal's triangle.

Sample Input

```
6
```

Sample Output

```
           1
         1   1
       1   2   1
     1   3   3   1
```

Sample Input

```
4
```

Sample Output

```
       1
     1   1
   1   2   1
 1   3   3   1
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q7. **Problem statement:**
Write a java program to print Floyd's Triangle.

Input Format

Input consists of 2 integer numbers.

Output Format

Refer to the sample output for the formatting specifications.

Sample Input

```
4
1
```

Sample Output

```
1
2  3
4  5  6
7  8  9  10
```

Sample Input

Sample Output

```
3
2
```

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

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q8. **Problem statement:**
Write a java program to program for half diamond pattern printing using numbers and stars.

Input Format

The input gets the N value.

Output Format

Refer to the sample output for the specifications.

Sample Input

```
4
```

Sample Output

```
1
2*2
3*3*3
4*4*4*4
```

Sample Input

```
3
```

Sample Output

```
1
2*2
3*3*3
3*3*3
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q9. **Problem statement:**
Write a java program to print a palindrome pyramid patterns using numbers.

Input Format

The first input consists of the N value.

Output Format

Refer to the sample output for the pattern to be printed.

Sample Input

```
5
```

Sample Output

```
1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Sample Input

```
4
```

Sample Output

```
1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q10. **Problem statement:**
Write a java program to palindrome pyramid pattern using numbers and stars.

Input Format

The first input consists of the N value.

Output Format

Refer to the sample output for the pattern to be printed.

Sample Input

```
7
```

Sample Output

```
********1********
*******2*2*******
******3*3*3******
*****4*4*4*4*****
```

Sample Input

```
6
```

Sample Output

```
********1********
*******2*2*******
******3*3*3******
*****4*4*4*4*****
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Answer Key & Solution

Section 1 - CODING

Q1

Test Case

Input

```
3
```

Output

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

Weightage - 25

Input

```
2
```

Output

```
*
*  *
```

Weightage - 25

Input

```
1
```

Output

```
*
```

Weightage - 25

Input

```
7
```

Output

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

Weightage - 25

Sample Input

```
4
```

Sample Output

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

Sample Input

```
5
```

Sample Output

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

Solution

```
 import java.util.*;
class Main
{
    public static void main(String[] args)
    {

        int rows;
        Scanner s=  new Scanner(System.in);
        rows=s.nextInt();
        for (int i = 1; i <= rows; ++i)
        {  //Outer loop for rows

            for (int j = 1; j <= i; ++j)
            { //Inner loop for Col
                 System.out.print("* "); //Print *
            }
        System.out.println(); //New line
         }
    }
}
```

Q2

Test Case

Input

```
3
```

Output

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

Weightage - 25

Input

```
7
```

Output

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

Weightage - 25

Input

```
6
```

Output

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

Weightage - 25

Input

```
2
```

Output

```
* *
*
```

Weightage - 10

Input

```
1
```

Output

```
*
```

Weightage - 15

Sample Input

```
5
```

Sample Output

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

Sample Input

```
4
```

Sample Output

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

Solution

```
 import java.util.*;
class Main
{
    public static void main(String[] args)
    {

        int rows;
        Scanner s=  new Scanner(System.in);
        rows=s.nextInt();
         for(int i = rows; i >= 1; --i)
         {  //For Loop for Row
             for(int j = 1; j <= i; ++j)
             { //For Loop for Col
                 System.out.print("* "); //Prints *
            }
         System.out.println(); //Get to newline
         }
    }
}
```

Q3

Test Case

Input

```
4
```

Output

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

Weightage - 25

Input

```
3
```

Output

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

Weightage - 25

Input

```
2
```

Output

```
 *
**
```

Weightage - 25

Input

```
1
```

Output

```
*
```

Weightage - 25

Sample Input

```
5
```

Sample Output

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

Sample Input

```
4
```

Sample Output

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

Solution

```
 import java.util.*;
class Main
{
    public static void main(String[] args)
    {

        int n;
        int i,j;
        Scanner s=  new Scanner(System.in);
        n=s.nextInt();

        for( i=1;i<=n;i++)
        {
            for( j=i;j<=n;j++)
            {
                System.out.print(" ");
            }
            for( j=1;j<=i;j++)
            {
                System.out.print("*");
            }
            System.out.println();
        }
    }
}
```

Q4 **Test Case**

Input

```
4
```

Output

```
1
1  2
1  2  3
1  2  3  4
```

Weightage - 25

Input

```
6
```

Output

```
1
1  2
1  2  3
1  2  3  4
```

Weightage - 25

Input

```
3
```

Output

```
1
1 2
1 2 3
```

Weightage - 25

Input

```
2
```

Output

```
1
1 2
```

Weightage - 25

Sample Input

```
5
```

Sample Output

```
1
1 2
1 2 3
1 2 3 4
```

Sample Input

```
4
```

Sample Output

```
1
1 2
1 2 3
1 2 3 4
```

Solution

```
 import java.util.*;
class Main
{
    public static void main(String[] args)
    {

        int n;
        int i,j;
        Scanner s=  new Scanner(System.in);
        n=s.nextInt();
        for ( i = 1; i <= n; ++i)
        {
            for ( j = 1; j <= i; ++j)
            {
                System.out.print(j + " ");
            }
      System.out.println();
    }
  }
}
```

Q5 **Test Case**

Input

Output

```
6
```

```
A
B  B
C  C  C
D  D  D  D
```

Weightage - 25

Input

```
3
```

Output

```
A
B  B
C  C  C
```

Weightage - 25

Input

```
4
```

Output

```
A
B  B
C  C  C
D  D  D  D
```

Weightage - 25

Input

```
8
```

Output

```
A
B  B
C  C  C
D  D  D  D
```

Weightage - 25

Sample Input

```
5
```

Sample Output

```
A
B  B
C  C  C
D  D  D  D
```

Solution

```
 import java.util.*;
class Main
{

  public static void main(String[] args)
  {
      Scanner sc = new Scanner(System.in);
      int last = Integer.parseInt(sc.nextLine());

    char  alphabet = 'A';

    for (int i = 1; i <= last; ++i)
    {
      for (int j = 1; j <= i; ++j)
      {
        System.out.print(alphabet + " ");
      }
      ++alphabet;

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

Q6

Test Case

Input

```
5
```

Output

```
                 1
              1     1
           1     2     1
        1     3     3     1
```

Weightage - 25

Input

```
5
```

Output

```
                 1
              1     1
           1     2     1
        1     3     3     1
```

Weightage - 25

Input

```
7
```

Output

```
                       1
                    1     1
                 1     2     1
              1     3     3     1
```

Weightage - 25

Input

```
3
```

Output

```
           1
        1     1
     1     2     1
```

Weightage - 25

Sample Input

```
6
```

Sample Output

```
                    1
                 1     1
              1     2     1
           1     3     3     1
```

Sample Input

```
4
```

Sample Output

```
              1
           1     1
        1     2     1
     1     3     3     1
```

Solution

```
import java.util.*;
class Main
{

  public static void main(String[] args)
  {
    int rows,coef=1;
    Scanner s = new Scanner(System.in);
    rows=s.nextInt();
    // coef=s.nextInt();

    for(int i = 0; i < rows; i++)
    {
      for(int space = 1; space < rows - i; ++space)
      {
        System.out.print("  ");
      }

      for(int j = 0; j <= i; j++)
      {
        if (j == 0 || i == 0)
          coef = 1;
        else
          coef = coef * (i - j + 1) / j;

        System.out.printf("%4d", coef);
      }

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

Q7

Test Case

Input

```
4
1
```

Output

```
1
2  3
4  5  6
7  8  9  10
```

Weightage - 25

Input

```
6
2
```

Output

```
2
3  4
5  6  7
8  9  10  11
```

Weightage - 25

Input

```
5
3
```

Output

```
3
4  5
6  7  8
9  10  11  12
```

Weightage - 25

Input

```
3
1
```

Output

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

Weightage - 25

Sample Input

```
4
1
```

Sample Output

```
1
2 3
4 5 6
7 8 9 10
```

Sample Input

```
3
2
```

Sample Output

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

Solution

```
import java.util.*;
class Main
{

  public static void main(String[] args)
  {
    int rows,number;
    Scanner s = new Scanner(System.in);
    rows=s.nextInt();
    number=s.nextInt();
    for(int i = 1; i <= rows; i++)
    {

      for(int j = 1; j <= i; j++)
      {
        System.out.print(number + " ");
        ++number;
      }
        System.out.println();
    }
  }
}
```

Q8

Test Case

Input

```
5
```

Output

```
1
2*2
3*3*3
4*4*4*4
```

Weightage - 25

Input

```
6
```

Output

```
1
2*2
3*3*3
4*4*4*4
```

Weightage - 25

Input

```
3
```

Output

```
1
2*2
```

```
3*3*3
```

Weightage - 25

Input

```
7
```

Output

```
1
2*2
3*3*3
4*4*4*4
```

Weightage - 25

Sample Input

```
4
```

Sample Output

```
1
2*2
3*3*3
4*4*4*4
```

Sample Input

```
3
```

Sample Output

```
1
2*2
3*3*3
3*3*3
```

Solution

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

Q9

Test Case

Input

```
7
```

Output

```
1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Weightage - 25

Input

```
6
```

Output

```
1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Weightage - 25

Input

```
3
```

Output

```
1
1 2 1
1 2 3 2 1
```

Weightage - 25

Input

```
2
```

Output

```
1
1 2 1
```

Weightage - 25

Sample Input

```
5
```

Sample Output

```
1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Sample Input

```
4
```

Sample Output

```
1
1 2 1
1 2 3 2 1
1 2 3 4 3 2 1
```

Solution

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

Q10

Test Case

Input

```
5
```

Output

```
********1********
*******2*2*******
******3*3*3******
*****4*4*4*4*****
```

Weightage - 25

Input

```
4
```

Output

```
********1********
*******2*2*******
******3*3*3******
*****4*4*4*4*****
```

Weightage - 25

Input

```
3
```

Output

```
********1********
*******2*2*******
******3*3*3******
```

Weightage - 25

Input

```
2
```

Output

```
********1********
*******2*2*******
```

Weightage - 25

Sample Input

```
7
```

Sample Output

```
********1********
*******2*2*******
******3*3*3******
*****4*4*4*4*****
```

Sample Input

```
6
```

Sample Output

```
********1********
*******2*2*******
******3*3*3******
```

Solution

```
import java.util.Scanner;
class Main
{
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        int i, j, space, count = 1, num = 0, star = 8;
        int n = sc.nextInt();
        space = n;
        for (i = 1; i <= n; i++)
        {
            for (j = 1; j <= star; j++)
                if(i + j <= star + 1)
                    System.out.print("*");
                    num++;
            for (j = 1; j <= i; j++)
            {
                System.out.print(num);
                if (i > 1 && count < i)
                {
                        System.out.print("*");
                        count++;
                }
        }
        for (j = 1; j <= star; j++)
            if(i + n <= j + n)
                System.out.print("*");
                System.out.println();
        space--;
        count = 1;
        }
    }
}
```