1.Make a program that prints the sequence like the following example.

**Input**

This problem doesn't have input.

| **Input Sample** | **Output Sample** |
| --- | --- |
|  | I=1,J=7 I=1,J=6 I=1,J=5 I=3,J=9 I=3,J=8 I=3,J=7 ... I=9,J=15 I=9,J=14 I=9,J=13 |

2. Write the java code for a program that takes an input of an integer n.

n \* 2 lines should be printed by this program as per the example below.

If the input is 5, the output should be as follows:

1 1 1

1 2 2

2 4 8

2 5 9

3 9 27

3 10 28

4 16 64

4 17 65

5 25 125

5 26 126

3...Write a program that reads a number and print the Fibonacci number corresponding to this read number. Remember that the first two elements of the Fibonacci series are 0 and 1 and each next term is the sum of the two preceding it. Make sure your program works for all inputs and does not crash for inputs 1, 2.

**Input**Number of terms

**Output**Print the array containing the terms of the Fibonacci series.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 5 | Fib[0]=0 Fib[1]=1 Fib[2]=1  Fib[3]=2  Fib[4]=3 |

4. Given a non-empty array of ints, create a new array containing the elements from the original array that come after the last 4 in the original array. The original array will contain at least one 4. Note that it is valid in java to create an array of length 0.

([2, 4, 1, 2]) → [1, 2]  
([4, 1, 4, 2]) → [2]  
([4, 4, 1, 2, 3]) → [1, 2, 3]

5. For each multiple of 10 in the given array, change all the values following it to be that multiple of 10, until encountering another multiple of 10. So {2, 10, 3, 4, 20, 5} yields {2, 10, 10, 10, 20, 20}.

([2, 10, 3, 4, 20, 5]) → [2, 10, 10, 10, 20, 20]  
([10, 1, 20, 2]) → [10, 10, 20, 20]  
([10, 1, 9, 20]) → [10, 10, 10, 20]

6.Read an integer value corresponding to a person's age (in days) and print it in years, months and days, followed by its respective message “Year(s)”, “Month(s)”, “Day(s)”.

7. Print the following pattern of a rhombus:

Input:4

Output:

\*\*\*\*  
 \*\*\*\*

\*\*\*\*

\*\*\*\*

8. Read a number and make a program which puts this number in the first position of an array N[10]. In each subsequent position, put the double of the previous position. For example, if the input number is 1, the array numbers ​​must be 1,2,4,8, and so on.

**Input**

The input contains an integer number **V.**

**Output**

Print the stored number of each array position, in the form "N[**i**] =**X**", where **i** is the position of the array and **x** is the stored number at the position **i**. The first number for**X** is **V**.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 1 | N[0]=1 N[1]=2 N[2]=4 ... |

9. Make a program that prints the sequence like the following example.

## Input

This problem doesn't have input.

## Output

Print the sequence like the example below.

| **Input Sample** | **Output Sample** |
| --- | --- |
|  | I=1 J=60 I=4 J=55 I=7 J=50 ... I=? J=0 |

10.Generate the following sequence:

Input: 4

3,7,9,11,27,15,81,19

11. Write a program that reads two numbers X and Y (X < Y). After this, show a sequence of 1 to y, passing to the next line to each X numbers.

**Input**

The input contains two integer numbers X and Y.

**Output**

Each sequence must be printed in one line, with a blank space between each number, like the following example.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 3 99 | 1 2 3  4 5 6  ...  97 98 99 |

12. **pattern printing**

**Input:3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| \* | \* | \* | \* | \* |
|  | \* | \* | \* |  |
|  |  | \* |  |  |
|  | \* | \* | \* |  |
| \* | \* | \* | \* | \* |

Input:4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| \* | \* | \* | \* | \* | \* | \* |
|  | \* | \* | \* | \* | \* |  |
|  |  | \* | \* | \* |  |  |
|  |  |  | \* |  |  |  |
|  |  | \* | \* | \* |  |  |
|  | \* | \* | \* | \* | \* |  |
| \* | \* | \* | \* | \* | \* | \* |

13.

An array is a *palindrome* if it reads the same in both directions. For example,

3 5 7 2 4 is not a palindrome; however, the following one is. 4 2 6 2 4

Write a program that reads in an array and checks if it is a palindrome. If it is, then print ‘YES’ else ‘NO’

14. . In this problem you need to read 15 numbers and must put them into two different arrays: **par**if the number is even or **impar**if this number is odd. But  the size of each of the two arrrays is only 5 positions. So every time you fill one of two arrays, you must print the entire array to be able to use it again for the next numbers that are read. At the end, all remaining numbers of each one of these two arrays must be printed beginning with the odd array. Each array can be filled as many times as wanted.

| **Input Sample** | **Output Sample** |
| --- | --- |
| 1 3 4 -4 2 3 8 2 5 -7 54 76 789 23 98 | par[0]=4 par[1]=-4 par[2]=2 par[3]=8 par[4]=2 impar[0]=1 impar[1]=3 impar[2]=3 impar[3]=5 impar[4]=-7 impar[0]=789 impar[1]=23 par[0]=54 par[1]=76 par[2] = 98 |

15. Given an array of integer values, sort the even numbers first in ascending order , then the odd numbers in ascending order. The even numbers should appear before the odd numbers. Note that, the sorted numbers must be in one single array, which should be printed at the end.

Input array (given) {6,11,3,2,8,9} Resulting array : {2,6,8,3,9,11}