Test Paper

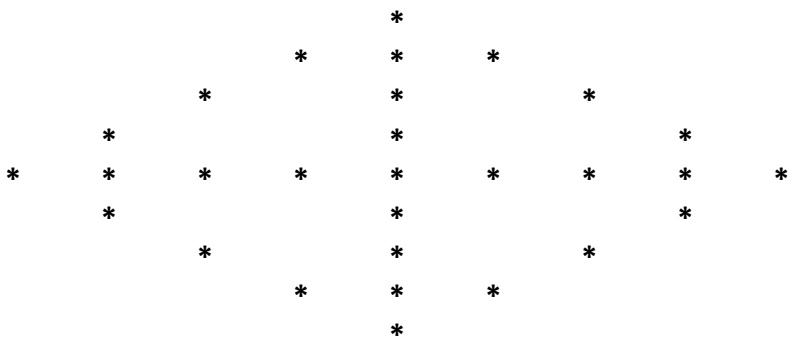
Q1. Write a java program to display 1 to nth Strong Number.

Q2. Write a Java program to display the following series:

100 81 64 49 36 25 16 9 4 1

(Perfect squares in reverse order)

Q3. Write a java program to print this pattern.



Q4. Write a java program to print this pattern.

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

Q5. Given a sorted array (may contain duplicates), find the first and last index of a given number x using binary search.

Example:

arr = [2, 4, 4, 4, 6, 7, 9], x = 4

Output: First = 1, Last = 3

Explanation :

- **Modify binary search:**
    - **To find first occurrence, if arr[mid] == x, move left to check more occurrences.**
    - **To find last occurrence, if arr[mid] == x, move right to check more occurrences.**
- **This requires two binary searches.**

Q6. Given a sorted array rotated at some pivot (like [6, 7, 1, 2, 3, 4, 5]), search for a number x using binary search.

Example:
arr = [6, 7, 1, 2, 3, 4, 5], x = 3
Output: Found at index 4
Explanation:

- **In rotated arrays, one half (left or right) is always sorted.**
- **Check which half is sorted:**
  - **If arr[left] <= arr[mid], left part is sorted.**
  - **Else, right part is sorted.**
- **Decide which half to discard based on x.**

Q7. Write a java program to count how many prime numbers are present in an ArrayList.

Explanation
For each number:

- **Check divisibility from 2 to n-1**
- **If divisible → not prime**
- **Count primes**

Q8. Write a Java program to store integer elements in a Vector and calculate the sum of all elements.

Input :- 10 20 30 40
Output :- Sum of Vector elements: 100

Q9. Longest Substring Without Repeating Characters

Description:
Find length of longest substring with unique characters.
Example:
Input: "abcabcbb"
Output: 3
Approach (HashSet + Sliding Window):

- **Maintain window with HashSet**
- **Expand right, shrink left on duplicates**
- **Track max length**

Q10. You are given an array of k linked-lists lists, each linked-list is sorted in ascending order.
Merge all the linked-lists into one sorted linked-list and return it.

Example 1:
Input: lists = [[1,4,5],[1,3,4],[2,6]]
Output: [1,1,2,3,4,4,5,6]
Explanation: The linked-lists are:
[
1->4->5,
1->3->4,
2->6
]

merging them into one sorted linked list:
1->1->2->3->4->4->5->6