Day-7

1. https://leetcode.com/problems/count-and-say/
   nahi bna

2. https://practice.geeksforgeeks.org/problems/longest-palindrome-in-a-string/0
->nahi bnta
3.https://practice.geeksforgeeks.org/problems/longest-repeating-subsequence2004/1

4. https://www.geeksforgeeks.org/print-subsequences-string/
Nahi hua
5. https://practice.geeksforgeeks.org/problems/permutations-of-a-given-string2041/1
Nahi bna
6. https://practice.geeksforgeeks.org/problems/median-in-a-row-wise-sorted-matrix1527/1#

```
class Solution{

public:

    int median(vector<vector<int>> &matrix, int r, int c){

        // code here

        vector<int>res;

        for(int i=0;i<matrix.size();i++)

        {

            for(int j=0;j<matrix[i].size();j++)

            {

                res.push_back(matrix[i][j]);

            }

        }

        sort(res.begin(),res.end());

        int len=res.size();

        if(len%2==1)

        {

            len=len/2+1;

        }

        else
        len/=2;

        return res[len-1];


    }

};
```

7. https://leetcode.com/problems/search-in-rotated-sorted-array/

```
class Solution {
public:
    int search(vector<int>& nums, int target) {
      if (nums.size() == 0) return -1;

      int left = 0, right = nums.size()-1;
       int start = 0;
       //1. find index of the smallest element
       while(left < right) {
           int mid = left + (right-left)/2;
           if (nums[mid] > nums[right]) {
               left = mid +1;
           } else right = mid;
       }

     //2. figure out in which side our target lies
       start = left;
       left = 0;
       right = nums.size()-1;
       if (target >= nums[start] && target <= nums[right])
           left = start;
       else right = start;

     //3. Run normal binary search in sorted half.
       while(left <= right) {
           int mid = left + (right - left)/2;
           if (nums[mid] == target) return mid;

           if (nums[mid] > target) right = mid-1;
           else left = mid + 1;
       }

       return -1;
    }
};
```