

# K Sai Dithvika CSE

## Day-2

### Anagram Problem

python

```
1 #User function Template for python3
2 class Solution:
3     def areAnagram(ob, s1, s2):
4         # code here
5
6         if sorted(s1)==sorted(s2):
7             return 1
8         return 0
9
10
```

C++

```
// } Driver Code Ends
// User function Template for C++
class Solution {
public:
    int areAnagram(string s1, string s2) {
        // code here
        sort(s1.begin(),s1.end());
        sort(s2.begin(),s2.end());
        if(s1==s2){
            return 1;
        }
        return 0;
    }
};
// } Driver Code Ends
```

### Row with max 1's

```

class Solution:
    def rowWithMax1s(self, arr):
        # code here
        n = len(arr)
        max row = -1
        max count = 0
        for i in range(n):
            count=arr[i].count(1)
            if count>max count:
                max count=count
                max row = i

        return max row

```

← } Driver Code Ends

## Longest consecutive subsequence

```

class Solution:
    # arr[] : the input array


    def findLongestConseqSubseq(self,arr):
        s=set(arr)
        mx in arr = max(arr)
        mx=1
        i=0
        while i<=mx in arr:
            temp=0
            num=i
            while num in s:
                temp+=1
                num+=1
            mx=max(mx,temp)
            i=num+1
        return mx

```

← } Driver Code Ends

## longest palindrome in a string

```
def LongestPalindrome(s):  
    def grow(x, y):  
        while x >= 0 and y < len(s) and s[x] == s[y]:  
            x -= 1  
            y += 1  
        return s[x + 1:y]  
  
    longest = ""  
    for idx in range(len(s)):  
        part1 = grow(idx, idx)  
        part2 = grow(idx, idx + 1)  
  
        if len(part1) > len(longest):  
            longest = part1  
        if len(part2) > len(longest):  
            longest = part2  
  
    return longest
```

 } Driver Code Ends