Problem Solving - 1

Thursday, 20 July 2023

1st Mock at 5:00 PM

Q1 Given array [1, 2, 3, 4, 5] and [1, 2, 3, 4, 5] find the number of pairs (i, j) such that arr[i] == arr[j]

→ Complex Name: Easy/medium

Q2 There is a sign post [1, 2, 3, 4, 5] [1, 2, 3, 4, 5] you are given a string of length n. you are given half down of the length can be '1' or '?' you need to build the sign post so that the total value of four

→ Complex + Tricky

Q3 Amplitude of an array: max-min in array. you are given an array and a number k. find the minimum amplitude you can get if you are allowed to change k elements at most

→ Complex - Tricky

Q4 Given a string S. Number of swaps to sort S is 2 swaps if 2 and that S[1:2:3] and sorted characters in S are distinct

→ Complex - Tricky

d d

S_1 freq[26] = {0,0 ... 0}; $d_1 = 0$

S_2 freq[26] = {0 ... 0}

for (int i=0; i<n; i++) {
freq[S[i]-'a']++;
if (freq[S[i]-'a'] == 1) { d_2++; }
}

S_2

for (i=0; i<N; i++) {
freq[S[i]-'a']--;
if (freq[S[i]-'a'] == 0) d_1--;
freq[S[i]-'a']++;
if (freq[S[i]-'a'] == 1) d_1++;
if (d_1 == d_2) cnt++;
}

S_1 S_2

$d_1 = 3$ (a)

S_2 a = ✗ b ✗ c ✗ 1, b = ✗ ✗ 0, c = 1

S_1 a = ✗ ✗ 3, b = ✗ ✗ 2, c = 0

"aabbac"

Q ⇒ you are given a number N. for every i ∈ [0, N] print the number of set bits in i. → __builtin_popcount(i) ✗

⇒ N = 5

0 1 2 3 4 5
0 1 1 2 1 2

1011010110001
1011010110000

6 ⇒ 1100 > 12

1100 < 1

n&1 → count

n/2 → n>>=1

n&=n-1 → n

n & n-1

1011010100000
1100000000000

(n-1)

Q ⇒ N log(N)

for (int i=0; i<=n; i++) {
ans += count(i)
}

count(i) {
cnt=0
while (i>0) {
cnt++
i &= i&(i-1);
}
return cnt;
}

O(set bits)

count(i) {
cnt=0;
while (i>0) {
if (i&1) cnt++
i = i/2
}
return cnt;
}

i/2

O(log(i))

count(i) ⇒ set bits in i

1011011010 0

count[n] ⇒
count[0] = 0;
for (i=1; i<=n; i++) {
if ((i%2 == 0)) count[i] = count[i/2]
else count[i] = count[i/2] + 1
}

count[i] = count[i/2] + (i&1);

⇒ you are given a string S (binary); you want to divide it in three parts, such that decimal representation of every part is equal

⇒ 0 0 1 0 0 0 0 0 1 0 0 1 0 0 0

N = 3

n%3 | n/3

6

2 | 2 | 2

00110 = 12 = 12