

Count Symmetric Integers

2843. Count Symmetric Integers

Solved

Easy Topics Companies Hint

You are given two positive integers `low` and `high`.

An integer x consisting of $2 \times n$ digits is **symmetric** if the sum of the first n digits of x is equal to the sum of the last n digits of x . Numbers with an odd number of digits are never symmetric.

Return the **number of symmetric** integers in the range `[low, high]`.

Example 1:

Input: `low = 1, high = 100`

Output: 9

Explanation: There are 9 symmetric integers between 1 and 100: 11, 22, 33, 44, 55, 66, 77, 88, and 99.

Example 2:

Input: `low = 1200, high = 1230`

Output: 4

Explanation: There are 4 symmetric integers between 1200 and 1230: 1203, 1212, 1221, and 1230.

Constraints:

- $1 \leq \text{low} \leq \text{high} \leq 10^4$

To find the number of digits in Integer x :-

$$\text{int } n0 = (\text{int})(\log_{10}(x) + 1)$$

```

C++ v Auto
1 class Solution {
2 public:
3     int countSymmetricIntegers(int low, int high) {
4         int ans=0;
5         for(int i=low;i<=high;i++){if(isSymmetric(i))ans++;
6         return ans;
7     }
8     bool isSymmetric(int x){
9         int y=(int)(log10(x)+1);
10        if((y&1)==1)return 0;
11        int p=y/2;
12        int p1=0, p2=0;
13        while(p){
14            int pp=x%10;
15            p1+=pp;
16            x/=10;
17            p--;
18        }
19        p=y/2;
20        while(p){
21            int pp=x%10;
22            p2+=pp;
23            x/=10;
24            p--;
25        }
26        return p1==p2;
27    }
28 };

```

T.C. $O(n \log n)$

S.C. $O(1)$

Constraints :-

- $1 \leq \text{low}, \text{high} \leq 10^4$

Approach 2 : Enumeration :-

Intuition :-

Enumerate all numbers from `low` to `high` :

- If it is a two-digit number & is a multiple of 11, then it is a symmetric integer.
- If it is a four-digit number, calculate the sum of the thousands & hundred digits, as well as the sum of the tens & ones digit. If these are equal, it is a symmetric (even)

well as the sum of the tens & ones digit.
If they are equal, it is a symmetric (even) integer.

Finally, it returns the number of symmetric integers in the range.

```
Code
C++ Auto
1 class Solution {
2 public:
3     int countSymmetricIntegers(int low, int high) {
4         int res=0;
5         for(int a=low; a<=high; a++){
6             if(a<100 && a%11==0){
7                 res++;
8             }
9             else if(1000<=a && a<10000){
10                 int left=a/1000 +(a%1000)/100;
11                 int right= (a%100)/10 +a%10;
12                 if(left == right)res++;
13             }
14         }
15         return res;
16     }
17 };
```

T.C : $O(\text{high} - \text{low})$

S.C : $O(1)$