Find Numbers with Even Number of Digits

03 July 2022   06:39

Given an array nums of integers, return how many of them contain an **even number** of digits.

From <https://leetcode.com/explore/learn/card/fun-with-arrays/521/introduction/3237/>

Example 1:
Input: nums = [12,345,2,6,7896]
Output: 2
Explanation:
12 contains 2 digits (even number of digits).
345 contains 3 digits (odd number of digits).
2 contains 1 digit (odd number of digits).
6 contains 1 digit (odd number of digits).
7896 contains 4 digits (even number of digits).
Therefore only 12 and 7896 contain an even number of digits.
Example 2:
Input: nums = [555,901,482,1771]
Output: 1
Explanation:
Only 1771 contains an even number of digits.

<u>Best</u>

```cpp
class Solution {
public:
    int findNumbers(vector<int>& nums)
    {
        int n=nums.size(),count=0;
        for(int i=0;i<n;i++){
            int res=log10(nums[i])+1;
            if(res%2==0){
                count++;
            }
        }
        return count;
    }
}
```

We are using log10 Maths function
->log10(x) return +ve number always
condition is that x must lie between[1,∞]
->Don't use double or float they will give output in decimals

<u>Solution2:</u>

```cpp
class Solution {

public:

int findNumbers(vector& nums) {

int count = 0; // final number to be returned as answer

    for(int i=0 ; i<nums.size() ; i++){
        int temp = 0;  // for checking even digits of a particular element
                            };

while(nums[i]>0){
        temp++;
        nums[i] = nums[i]/10;
    }

    if(temp%2 == 0){
        count++;
    }
  }

  return count;
}

};
```

<u>Solution 3:</u>

```cpp
int findNumbers(vector& nums) {
int count=0;
for(auto e : nums) if(to_string(e).length()%2==0) count++;
return count;
}
```
//Converting numbers into string and using length() getting length of the string