**Find Numbers with Even Number of Digits**

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

**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.

**Constraints:**

* 1 <= nums.length <= 500
* 1 <= nums[i] <= 105

**Solution :**

**class Solution {**

**public:**

**int findNumbers(vector<int>& nums) {**

**int n = nums.size();**

**int res=0;**

**for(int i=0; i<n; i++){**

**int count=0;**

**while(nums[i]>0){**

**count++;**

**nums[i]=nums[i]/10;**

**}**

**if(count%2==0)**

**res++;**

**}**

**return res;**

**}**

**};**