Even Digits

# Question

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.

# Pseudo Code

Initialize the digitCounter, evenDigitCounter and tempValue to 0

Run the Outer For loop

tempValue <- first element of the Array

Set the value of digitCounter to 0

Run the while loop with condition tempValue / 10 != 0

Increment the digitCounter by One

tempValue <- tempValue / 10

If (digitCounter + 1) is Even

Increment the Value of evenDigitCounter by One.

Return evenDigitCounter

# Source Code

## V 1.0

1. int findNumbers(int\* nums, int numsSize){
3. int digitCounter = 0, evenDigitCounter = 0, tempValue = 0;
5. for(int i=0 ; i<numsSize ; i++) {
6. tempValue = \*(nums + i);
7. digitCounter = 0;
9. while((tempValue / 10) != 0) {
10. digitCounter++;
11. tempValue = tempValue / 10;
12. }
14. if((digitCounter + 1) % 2 == 0) {
15. evenDigitCounter++;
16. }
17. }
19. return evenDigitCounter;
20. }