Problem 2094: Finding 3-Digit Even Numbers

Problem Information

Difficulty: Easy
Acceptance Rate: 78.87%
Paid Only: No
Tags: Array, Hash Table, Recursion, Sorting, Enumeration

Problem Description

You are given an integer array `digits`, where each element is a digit. The array may contain duplicates.

You need to find **all** the **unique** integers that follow the given requirements:

* The integer consists of the **concatenation** of **three** elements from `digits` in **any** arbitrary order. * The integer does not have **leading zeros**. * The integer is **even**.

For example, if the given `digits` were `[1, 2, 3]`, integers `132` and `312` follow the requirements.

Return _a**sorted** array of the unique integers._

Example 1:

Input: digits = [2,1,3,0] **Output:** [102,120,130,132,210,230,302,310,312,320] **Explanation:** All the possible integers that follow the requirements are in the output array. Notice that there are no **odd** integers or integers with **leading zeros**.

Example 2:

Input: digits = [2,2,8,8,2] **Output:** [222,228,282,288,822,828,882] **Explanation:** The same digit can be used as many times as it appears in digits. In this example, the digit 8 is used twice each time in 288, 828, and 882.

Example 3:

Input: digits = [3,7,5] **Output:** [] **Explanation:** No **even** integers can be formed using the given digits.

Constraints:

* `3 <= digits.length <= 100` * `0 <= digits[i] <= 9`

Code Snippets

C++:

```
class Solution {
public:
vector<int> findEvenNumbers(vector<int>& digits) {

}
};
```

Java:

```
class Solution {
public int[] findEvenNumbers(int[] digits) {

}
}
```

Python3:

```
class Solution:
def findEvenNumbers(self, digits: List[int]) -> List[int]:
```