# Problem 1999: Smallest Greater Multiple Made of Two Digits

## Problem Information

**Difficulty:** Medium
**Acceptance Rate:** 48.15%
**Paid Only:** Yes
**Tags:** Math, Enumeration

## Problem Description

Given three integers, `k`, `digit1`, and `digit2`, you want to find the **smallest** integer that is:

* **Larger** than `k`, * A **multiple** of `k`, and * Comprised of **only** the digits `digit1` and/or `digit2`.

Return _the**smallest** such integer. If no such integer exists or the integer exceeds the limit of a signed 32-bit integer (_`231 - 1` _), return_`-1`.

**Example 1:**

**Input:** k = 2, digit1 = 0, digit2 = 2 **Output:** 20 **Explanation:** 20 is the first integer larger than 2, a multiple of 2, and comprised of only the digits 0 and/or 2.

**Example 2:**

**Input:** k = 3, digit1 = 4, digit2 = 2 **Output:** 24 **Explanation:** 24 is the first integer larger than 3, a multiple of 3, and comprised of only the digits 4 and/or 2.

**Example 3:**

**Input:** k = 2, digit1 = 0, digit2 = 0 **Output:** -1 **Explanation:** No integer meets the requirements so return -1.

**Constraints:**

* `1 <= k <= 1000` * `0 <= digit1 <= 9` * `0 <= digit2 <= 9`

## Code Snippets

**C++:**

```
class Solution {
public:
int findInteger(int k, int digit1, int digit2) {

}
};
```

**Java:**

```
class Solution {
public int findInteger(int k, int digit1, int digit2) {

}
}
```

**Python3:**

```
class Solution:
def findInteger(self, k: int, digit1: int, digit2: int) -> int:
```