

Problem 728: Self Dividing Numbers

Problem Information

Difficulty: Easy

Acceptance Rate: 80.10%

Paid Only: No

Tags: Math

Problem Description

A **self-dividing number** is a number that is divisible by every digit it contains.

* For example, `128` is **a self-dividing number** because `128 % 1 == 0`, `128 % 2 == 0`, and `128 % 8 == 0`.

A **self-dividing number** is not allowed to contain the digit zero.

Given two integers `left` and `right`, return _a list of all the**self- dividing numbers** in the range_ `[left, right]` (both **inclusive**).

Example 1:

Input: left = 1, right = 22 **Output:** [1,2,3,4,5,6,7,8,9,11,12,15,22]

Example 2:

Input: left = 47, right = 85 **Output:** [48,55,66,77]

Constraints:

* `1 <= left <= right <= 104`

Code Snippets

C++:

```
class Solution {  
public:  
vector<int> selfDividingNumbers(int left, int right) {  
  
}  
};
```

Java:

```
class Solution {  
public List<Integer> selfDividingNumbers(int left, int right) {  
  
}  
}
```

Python3:

```
class Solution:  
def selfDividingNumbers(self, left: int, right: int) -> List[int]:
```