Problem 1215: Stepping Numbers

Problem Information

Difficulty: Medium
Acceptance Rate: 47.94%
Paid Only: Yes
Tags: Math, Backtracking, Breadth-First Search

Problem Description

A **stepping number** is an integer such that all of its adjacent digits have an absolute difference of exactly `1`.

* For example, `321` is a **stepping number** while `421` is not.

Given two integers `low` and `high`, return _a sorted list of all the**stepping numbers** in the inclusive range_ `[low, high]`.

Example 1:

Input: low = 0, high = 21 **Output:** [0,1,2,3,4,5,6,7,8,9,10,12,21]

Example 2:

Input: low = 10, high = 15 **Output:** [10,12]

Constraints:

* `0 <= low <= high <= 2 * 109`

Code Snippets

C++:

```
class Solution {
public:
vector<int> countSteppingNumbers(int low, int high) {

}
};
```

Java:

```
class Solution {
public List<Integer> countSteppingNumbers(int low, int high) {

}
}
```

Python3:

```
class Solution:
def countSteppingNumbers(self, low: int, high: int) -> List[int]:
```