Problem 1306: Jump Game III

Problem Information

Difficulty: Medium
Acceptance Rate: 66.50%
Paid Only: No
Tags: Array, Depth-First Search, Breadth-First Search

Problem Description

Given an array of non-negative integers `arr`, you are initially positioned at `start` index of the array. When you are at index `i`, you can jump to `i + arr[i]` or `i - arr[i]`, check if you can reach **any** index with value 0.

Notice that you can not jump outside of the array at any time.

Example 1:

Input: arr = [4,2,3,0,3,1,2], start = 5 **Output:** true **Explanation:** All possible ways to reach at index 3 with value 0 are: index 5 -> index 4 -> index 1 -> index 3 index 5 -> index 6 -> index 4 -> index 1 -> index 3

Example 2:

Input: arr = [4,2,3,0,3,1,2], start = 0 **Output:** true **Explanation:** One possible way to reach at index 3 with value 0 is: index 0 -> index 4 -> index 1 -> index 3

Example 3:

Input: arr = [3,0,2,1,2], start = 2 **Output:** false **Explanation:** There is no way to reach at index 1 with value 0.

Constraints:

* `1 <= arr.length <= 5 * 104` * `0 <= arr[i] < arr.length` * `0 <= start < arr.length`

Code Snippets

C++:

```
class Solution {
public:
bool canReach(vector<int>& arr, int start) {

}
};
```

Java:

```
class Solution {
public boolean canReach(int[] arr, int start) {

}
}
```

Python3:

```
class Solution:
def canReach(self, arr: List[int], start: int) -> bool:
```