Problem 1345: Jump Game IV

Problem Information

Difficulty: Hard
Acceptance Rate: 46.19%
Paid Only: No
Tags: Array, Hash Table, Breadth-First Search

Problem Description

Given an array of integers `arr`, you are initially positioned at the first index of the array.

In one step you can jump from index `i` to index:

* `i + 1` where: `i + 1 < arr.length`. * `i - 1` where: `i - 1 >= 0`. * `j` where: `arr[i] == arr[j]` and `i != j`.

Return _the minimum number of steps_ to reach the **last index** of the array.

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

Example 1:

Input: arr = [100,-23,-23,404,100,23,23,23,3,404] **Output:** 3 **Explanation:** You need three jumps from index 0 --> 4 --> 3 --> 9. Note that index 9 is the last index of the array.

Example 2:

Input: arr = [7] **Output:** 0 **Explanation:** Start index is the last index. You do not need to jump.

Example 3:

Input: arr = [7,6,9,6,9,6,9,7] **Output:** 1 **Explanation:** You can jump directly from index 0 to index 7 which is last index of the array.

Constraints:

* `1 <= arr.length <= 5 * 104` * `-108 <= arr[i] <= 108`

Code Snippets

C++:

```
class Solution {
public:
int minJumps(vector<int>& arr) {

}
};
```

Java:

```
class Solution {
public int minJumps(int[] arr) {

}
}
```

Python3:

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