} Daily Coding Problem

Daily Coding Problem #106

Problem

This problem was asked by Pinterest.

Given an integer list where each number represents the number of hops you can make, determine whether you can reach to the last index starting at index 0.

For example, `[2, 0, 1, 0]` returns `true` while `[1, 1, 0, 1]` returns `false`.

Solution

This problem can be solved by using dynamic programming:

We can see we can reach the Kth step if we can reach a step 0 `<= j < K` and `j + hops[j] >= K`.

Another approach that would use O(1) space is to start at the first step and keep a variable `steps_left` which represents the maximum number of steps we can move forward. If we store this value and ensure that we can reach the end of each step without this value becoming `0` (except for the last step), then we know we can reach the last step.

```python
def can_reach_end(hops):
    steps_left = 1

    for i in range(len(hops) - 1):
        steps_left = max(steps_left - 1, hops[i])
        if steps_left == 0:
            return False
    return True
```