Jumps Game II

For this problem we need the minimum number of jumps to reach the last index, guaranteed reachable.

Key Idea:
Greedy Level Traversal:
Think of the array as levels of BFS:
- farthest = farthest index reachable within current level.
- end = end of current level
- Each time we pass end, increment jumps and set end = farthest.

Algorithm:
1. Initialize:
   - jumps = 0
   - end = 0 (current level boundary)
   - farthest = 0 (max reachable index so far)
2. Iterate from i = 0 to n-2:
   - Update farthest = max (farthest, i + nums [i])
   - If i == end:
     - Increment jumps
     - Update end = farthest
3. Return jumps

Complexity:
- Time: $O(n)$ - single pass
- Space: $O(1)$ - constant extra space.