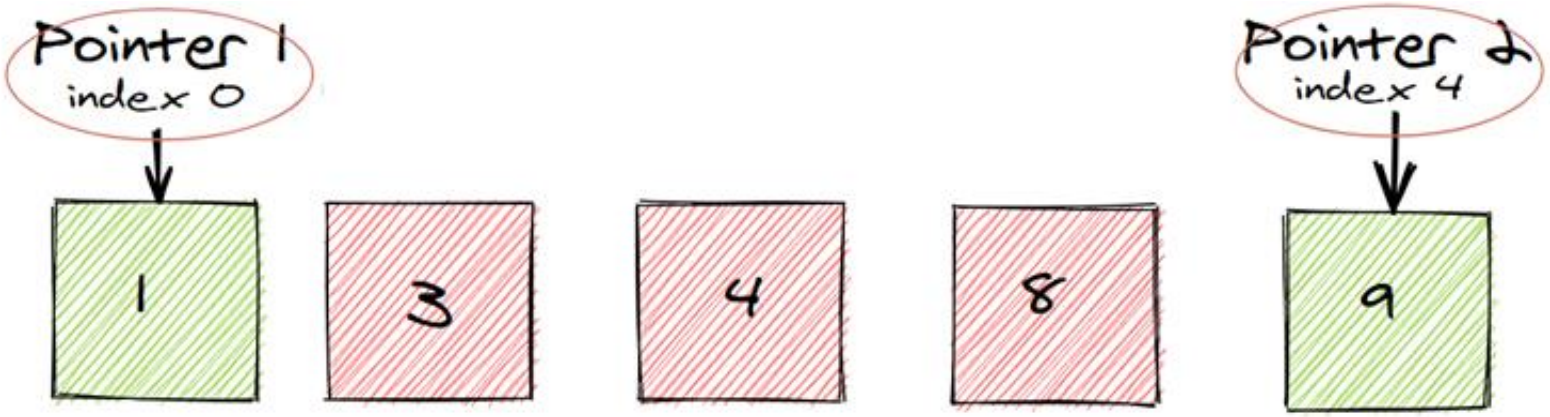
2 Pointer Approach

2 Pointer Technique

- ✓ Use 2 reference to move the control
- ✓ Usually used when solving Array or String problems
- ✓ Solves time and space complexity issue !!



Scenario

Reverse the Array

Input Array Sequence :

1	2	3	4	5

Working Steps

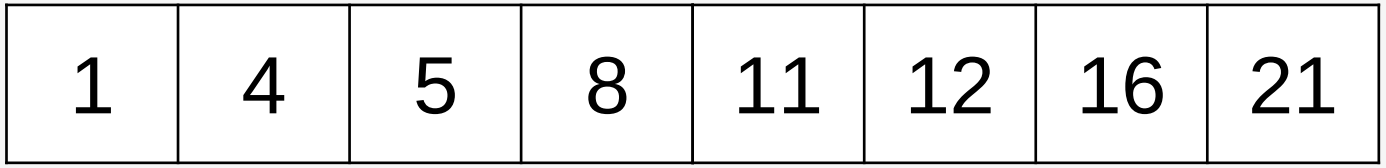
Scenario

Two Sum

Given an array of integers nums and an integer target, return indices of the two numbers such that they add up to target.

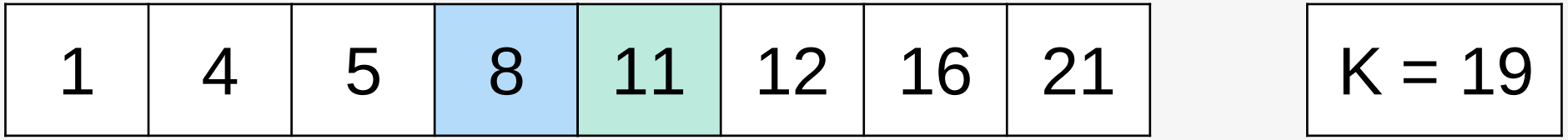
Input: nums = [1,4,5,8,11,12,16,21], target = 19
Output: [3,4]
Output: Because nums[3] + nums[4] == 19, we return [3, 4].

Two Pointer



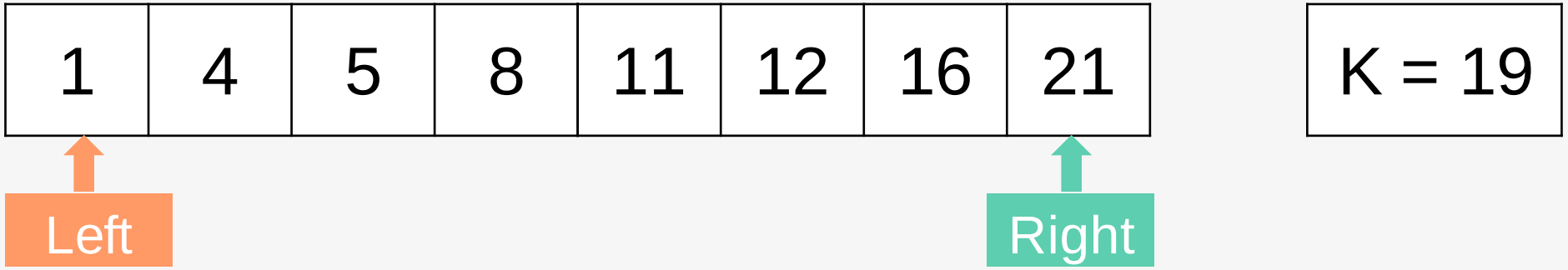
K = 19

Two Pointer



K = 19

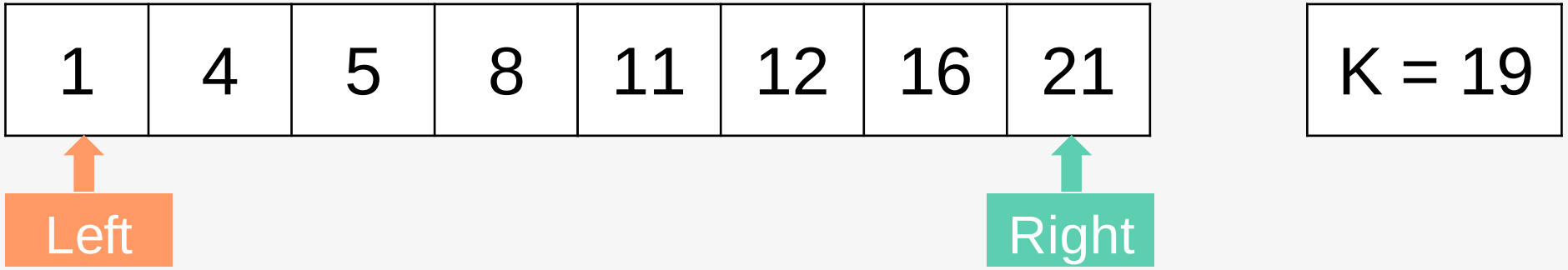
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

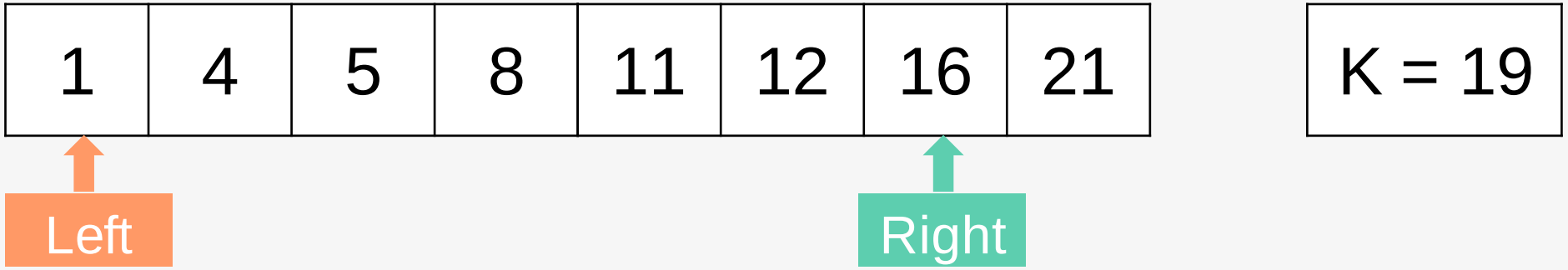
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

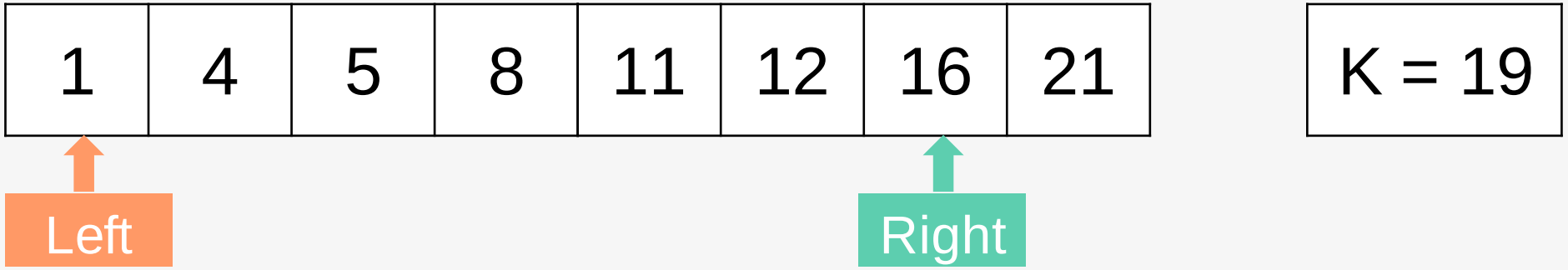
Two Pointer



K = 19

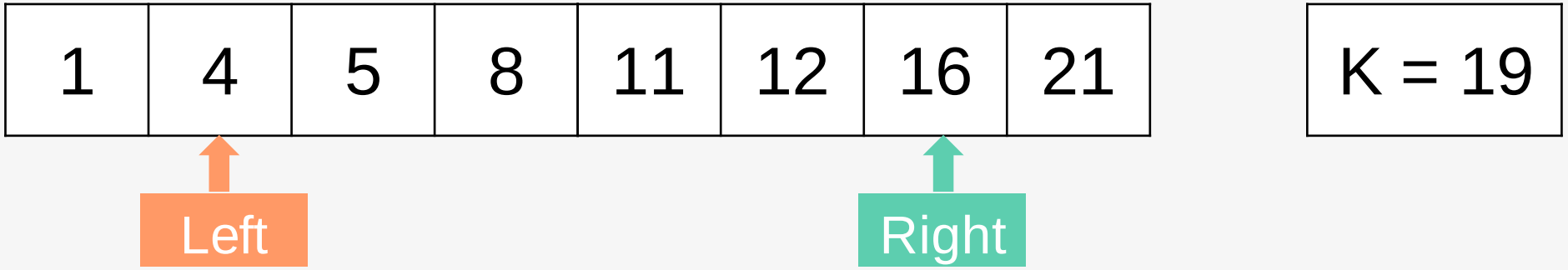
1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

Two Pointer



1) Left + Right = Sum (You got it, Break)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

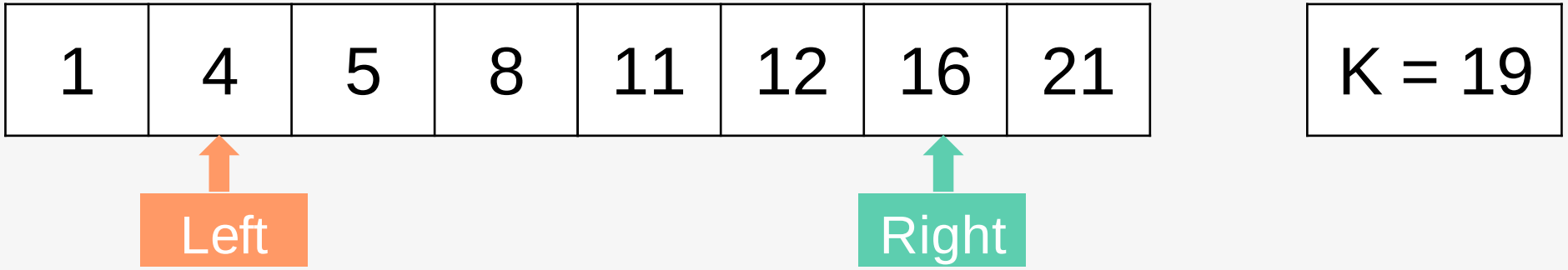
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

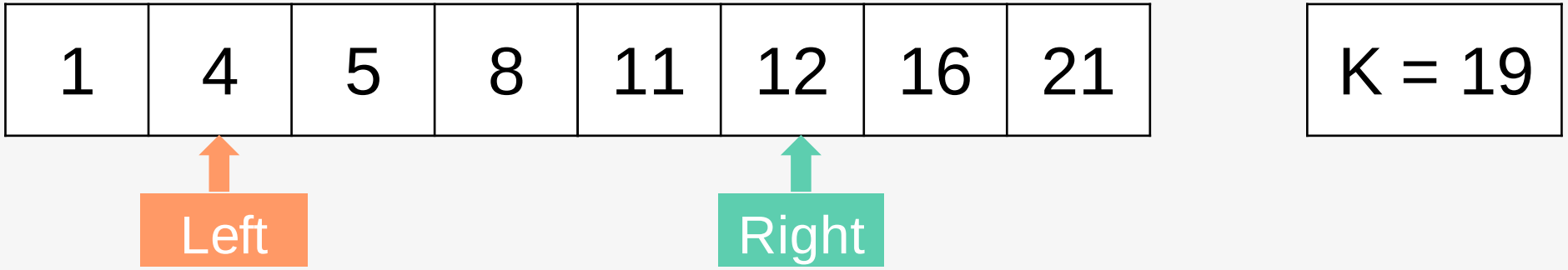
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

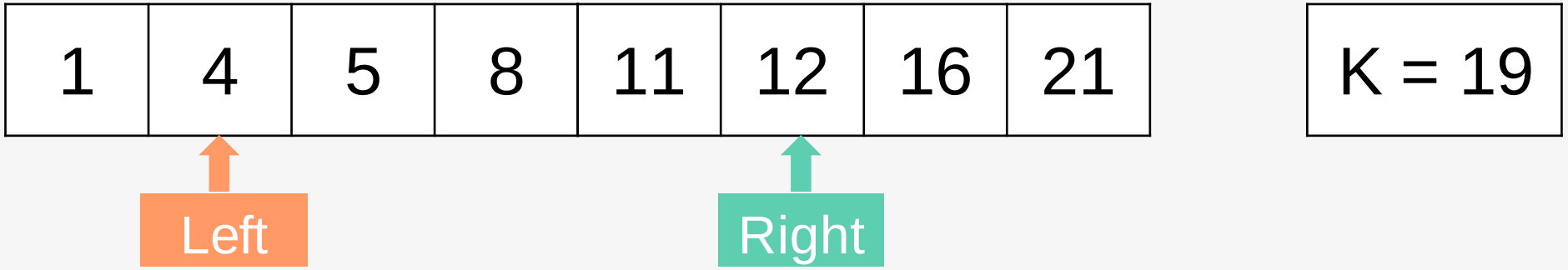
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

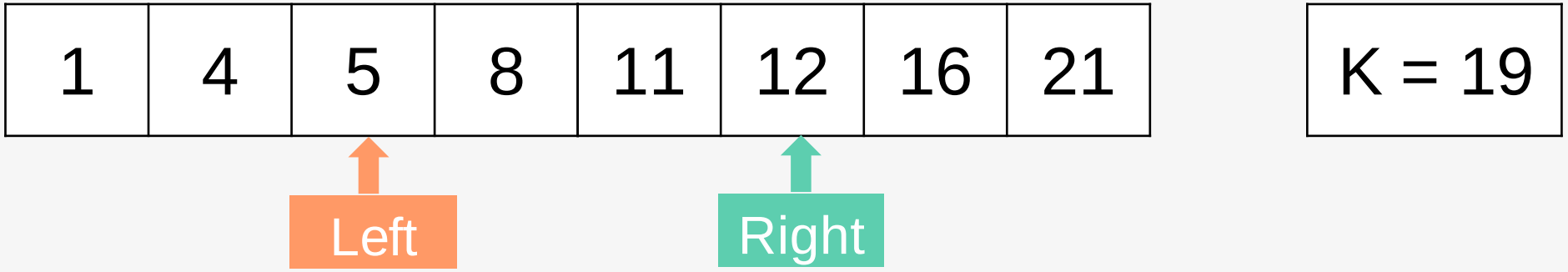
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

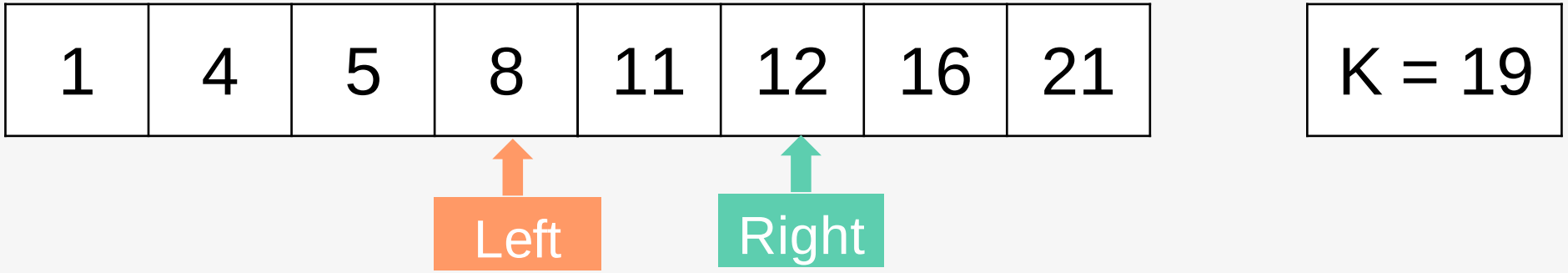
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

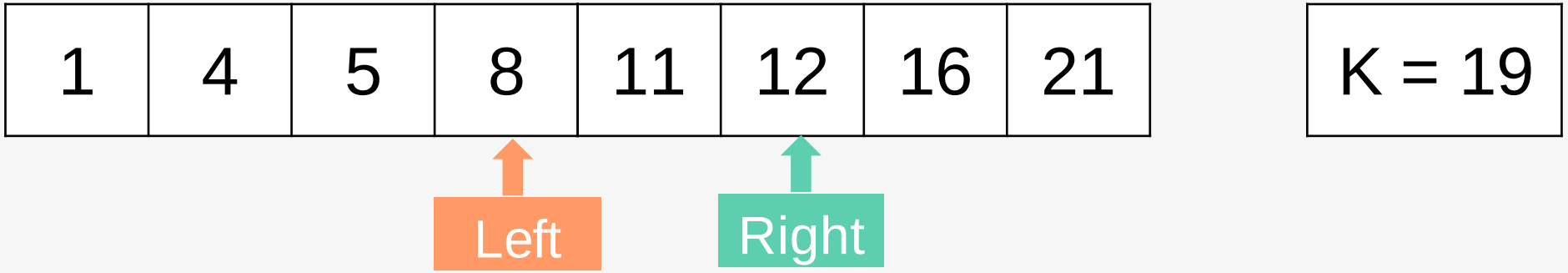
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

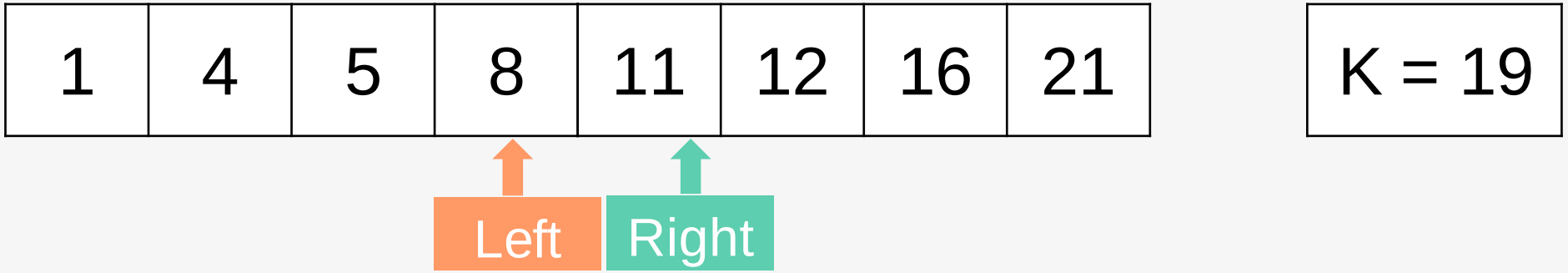
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

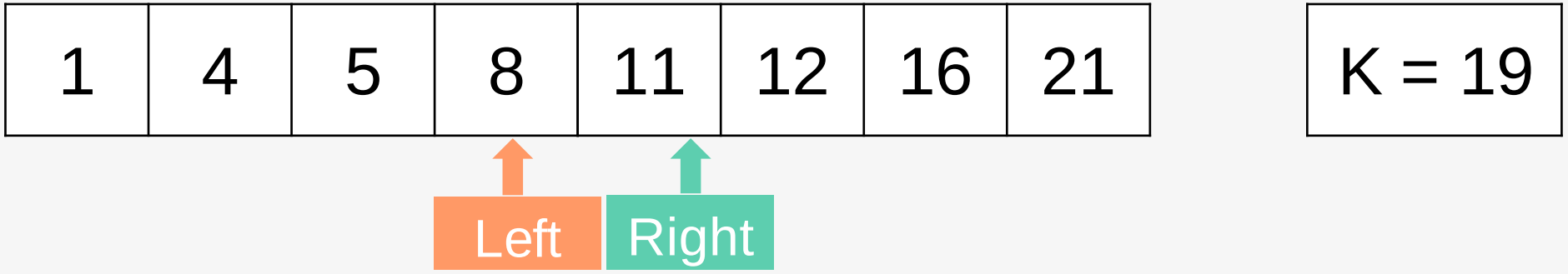
Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)

Two Pointer



K = 19

1) Left + Right = Sum (You got it, Return the indices)
2) Left + Right > Sum (Move Right towards Left Decrement Right)
3) Left + Right < Sum (Move Left towards Right Increment Left)