

 Ask a Question

Solution: 3Sum

Let's solve the 3Sum problem using the Two Pointers pattern.

We'll cover the following



- Statement
- Solution
 - Naive approach
 - Optimized approach using two pointers
 - Solution summary
 - Time complexity
 - Space complexity

?

Statement

Tt

Given an array of integers, `nums`, and an integer value, `target`, determine if there are any three integers in `nums` whose sum is equal to the `target`, that is, `nums[i] + nums[j] + nums[k] == target`. Return `TRUE` if



such integers exist in the array. Otherwise, return FALSE.



Note: A valid triplet consists of elements with *distinct* indexes. This means, for the triplet $\text{nums}[i]$, $\text{nums}[j]$, and $\text{nums}[k]$, $i \neq j$, $i \neq k$ and $j \neq k$.

Constraints:

- $3 \leq \text{nums.length} \leq 500$
- $-10^3 \leq \text{nums}[i] \leq 10^3$
- $-10^3 \leq \text{target} \leq 10^3$

