**Algorithm and Data Structures**

**Task**

Solve the following problem: Given an array of integers, return indices of the two numbers such that they add up to a specific target.

**Solution Approach**

To solve this problem, we can use a hash map (or dictionary) to efficiently find two indices where their corresponding values add up to the target. Here’s a step-by-step approach:

1. **Initialize a Hash Map**:
   * Use a hash map to store the indices of the numbers we have encountered so far.
2. **Iterate Through the Array**:
   * For each number in the array, compute its complement with respect to the target (i.e., target - current number).
   * Check if this complement exists in the hash map:
     + If it does, return the index of the complement (stored in the hash map) and the current index.
     + If it does not, add the current number and its index to the hash map.
3. **Return the Result**:
   * If the loop completes without finding such a pair, return an indication that no solution exists (though per problem constraints, a solution is guaranteed).

**Complexity Analysis**

* **Time Complexity**: O(n)
  + Each element is processed exactly once, and hash map operations (insertion and lookup) are average O(1).
* **Space Complexity**: O(n)
  + The space complexity is due to storing elements in the hash map.