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

You may assume that each input would have ***exactly* one solution**, and you may not use the *same* element twice.

You can return the answer in any order.

**Example 1:**

**Input:** nums = [2,7,11,15], target = 9

**Output:** [0,1]

**Explanation:** Because nums[0] + nums[1] == 9, we return [0, 1].

class Solution {

public:

    vector<int> twoSum(vector<int>& nums, int target) {

        unordered\_map<int,int> numMap;

        int n = nums.size();

        for(int i=0;i<n;i++){

            int comp=target - nums[i];

            if (numMap.count(comp)){

                return {numMap[comp],i};

            }

            numMap[nums[i]]=i;

        }

        return {};

    }

};