/\*

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

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

Example:

Given nums = [2, 7, 11, 15], target = 9,

Because nums[0] + nums[1] = 2 + 7 = 9,

return [0, 1].

由于一样的元素只能用一次，涉及查找，map肯定是最快的。map<int,int>存下标

\*/

class Solution {

public:

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

{

map<int,int> p;

for(int i=0;i<nums.size();i++)

{

p[nums[i]]=i;

}

vector<int> ret;

for(int i=0;i<nums.size();i++)

{

map<int,int>::iterator it=p.find(target-nums[i]);

if(it!=p.end() && i!=it->second)

{

ret.push\_back(i);

ret.push\_back(it->second);

return ret;

}

}

return ret;

}

};