# 题目

给你一个数字数组arr。

如果一个数列中，任意相邻两项的差总等于同一个常数，那么这个数列就称为 等差数列 。

如果可以重新排列数组形成等差数列，请返回true；否则，返回false。

**示例 1：**

输入：arr = [3,5,1]

输出：true

解释：对数组重新排序得到 [1,3,5] 或者 [5,3,1] ，任意相邻两项的差分别为 2 或 -2 ，可以形成等差数列。

**示例 2：**

输入：arr = [1,2,4]

输出：false

解释：无法通过重新排序得到等差数列。

**提示：**

2 <= arr.length <= 1000

-10^6 <= arr[i] <= 10^6

# 分析

## 方法一：数学公式

class Solution {

public:

bool canMakeArithmeticProgression(vector<int>& arr) {

std::sort(arr.begin(),arr.end());

for(int i=1;i<arr.size()-1;i++)

{

if(arr[i]\*2 != arr[i-1] + arr[i+1])

return false;

}

return true;

}

};

## 方法二：排序

**思路：**

先对原来数组排序，然后判断是否是等差数列，但这个排序是O(N²)。

**代码：**

class Solution {

public:

bool canMakeArithmeticProgression(vector<int>& arr) {

int sub=0;

bool rez=true;

for(int i=arr.size();i>0;i--){

for(int j=1;j<i;j++){

if(arr[j]>=arr[j-1]){

int temp=arr[j];

arr[j]=arr[j-1];

arr[j-1]=temp;

}

}

}

sub=arr[1]-arr[0];

for(int i=1;i<arr.size();i++){

if(arr[i]-arr[i-1]!=sub){

rez=false;

break;

}

}

return rez;

}

};