

23、三元组的最小距离/

24计算机考研成员一战成硕!



题目描述:

【2020统考真题】定义三元组(a, b, c)(a、b、c均为正数)的距离D=|a-b|+|b-c|+|c-a|给定3个非空整数集合S1、S2和S3,按升序分别存储在3个数组中。请设计一个尽可能高效的算法,计算并输出所有可能的三元组(a, b, c)(a \in S1,b \in S2,C \in S3)中的 最小距离。例如S1={-1,0,9},S2={-25,-10,10,11),S3={2,9,17,30,41},则最小距离为2,相应的三元组为(9,10,9)。要求:

要求:

- 1)给出算法的基本设计思想。
- 2) 根据设计思想,采用C或C++语言描述算法,关键之处给出注释。
- 3) 说明你所设计的算法的时间复杂度和空间复杂度。

☑ 倒计时

1、知识点及难度



解人: 多动症男孩

难度: 中上

知识点:

1. 本题的难度有所提高,可以优先考虑"暴力解法",即通过三个嵌套for循环是最简单的。

2. 解题思路中a代表的是当前遍历的三个数组元素中的最小值,c为最大值,不要混淆!!!

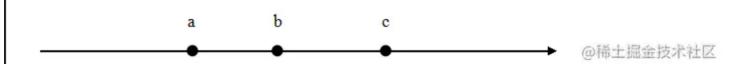
时间复杂度: *O(n+m+v)*---最坏情况下,访问了三个数组中的全部元素,其中n,m,v分别为三个数组的长度

空间复杂度: O(1)--- 数组为必要空间,其他辅助变量均为常数级,除此无额外的辅助空间

2、算法题

思路

本题的算法思想比较考验水平,所以这里尽量的详细一些,通过<mark>图文</mark>的方式,给大家好好解释一下。题目让我们求三元组中的最小距离,当然可以通过三个for循环实现暴力解法,既然可以通过嵌套循环实现,那一般也可以通过指针实现并降低时间复杂度,这里我们采用三指针的方式实现。假设i j k 分别指向三个数组 a b c,呢我们可以通过一个while循环来计算 |a[i]-b[j]| + |b[j]-c[k]| + |c[k]-a[i]|,关键点是在于每次循环后,该如何更新指针变量。这里详细解释一下:



如上图: a b c就相当于每次遍历的三个数组元素,三元组的距离自然是ab+bc+ac = 2ac(可以发现距离与点b没有关系)。我们要求的是最小距离,所以我们要考虑的就是如何移动这a c这两个点,使得ab+bc+ac的值最小。

- 1. 假设将a向右移动,可以发现ac变小,没问题,但向左移动,则ac变大,不可以
- 假设将c向右移动,可以发现ac变大,不可以,但向左移动,则ac变小,可以。 综上我们可以发现,有两种移动指针的方式,可以使得每次遍历得到的三元组距离变小,其中我们 采用第一种方式,将 a 向右移动。 (● 够详细了吧!!!)

作者:多动症男孩(原链接里代码可上机,有需要的同学可以自取)

原来的链接: https://juejin.cn/post/7211693149823434810

来源:稀土掘金

```
1 #include <stdio.h>
2
 3 // 求三数中的最小值
 4 int MinNum(int a, int b, int c) {
      int min = a < b ? a : b;</pre>
 5
      min = min < c? min : c;
 6
      return min;
7
8 }
10 // 求三元组的最小距离
11 int MinDistance(int a[], int b[], int c[], int aLen, int bLen, int cLen) {
      int minDis = 1000000000; // 初始值为无穷大,表示三元组中的最小距离
12
13
      int curDis = 0;
                             // 表示当前三元组距离
      int min = 0;
                             // 当前遍历的三个数中的最小值
14
      int i = 0, j = 0, k = 0; // 三个整型数组的指针
15
      while (i < aLen && j < bLen && k < cLen) {
16
          // 计算当前三元组距离
17
          curDis = abs(a[i] - b[j]) + abs(b[j] - a[i]) + abs(c[k] - a[i]);
18
          if (curDis < minDis) minDis = curDis;</pre>
19
20
          min = MinNum(a[i], b[j], c[k]);
          // 让当前最小的数哪一个数的数组指针向右(向前)移动一位
21
          if (min == a[i]) i++;
22
          else if (min == b[i]) i++;
23
24
          else if (min = c[k]) k++;
25
      }
      return minDis; // 返回结果
26
27 }
28
```

基本实现C++

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 // 求三数中的最小值
6 int MinNum(int a, int b, int c) {
7    int min = a < b ? a : b;
8    min = min < c? min : c;
9    return min;
10 }
11</pre>
```

```
12 // 求三元组的最小距离
13 int MinDistance(vector<int> a, vector<int> b, vector<int> c) {
      int minDis = 1000000000; // 初始值为无穷大,表示三元组中的最小距离
14
15
      int curDis = 0; // 表示当前三元组距离
      int min = 0;
                          // 当前遍历的三个数中的最小值
16
      int i = 0, j = 0, k = 0; // 三个整型数组的指针
17
      while (i < a.size() && j < b.size() && k < c.size()) {</pre>
18
          // 计算当前三元组距离
19
20
          curDis = abs(a[i] - b[j]) + abs(b[j] - a[i]) + abs(c[k] - a[i]);
         if (curDis < minDis) minDis = curDis;</pre>
21
          min = MinNum(a[i], b[j], c[k]);
22
         // 让当前最小的数哪一个数的数组指针向右(向前)移动一位
23
         if (min == a[i]) i++;
24
25
         else if (min == b[j]) j++;
        else if (min = c[k]) k++;
26
27
      }
      return minDis; // 返回结果
28
29 }
30
31
```

3、总结



蓝蓝B站首页:蓝蓝希望你上岸呀B站首页

蓝蓝公众号: 算法训练营9分计划

蓝蓝知识星球介绍: 图关于知识星球的权益

如何在星球打卡记录:

• 计算机考研数据结构算法专项day[1/60]:

• 学习内容: 最好能发出自己写的图片

• 遇到的问题:如果无就不用写了

• 小结:这部分一周写一次即可。