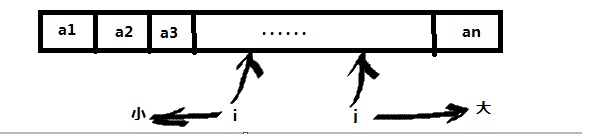
1. 问题描述

给定排的数组A，大小为n，现给定数X，判断A中是否存在两数之和等于X。给出一个O(nlg(n))的算法。

2. 解决思路

首先对数组进行排序，如果使用归并的排序的话，算法的复杂度在nlg(n)级别，然后通过从中间相两边查找即可。



3. 实现代码

复制代码

/// <summary>  
        /// 在给定的数组查找是否存在和为sum的两个元素，如果存在  
        /// 函数返回true，否则false  
        /// </summary>  
        /// <param name="arr"></param>  
        /// <param name="sum"></param>  
        static bool Find(int[] arr, int sum)  
        {  
            // 对数组排序  
            MergeSort(ref arr, 0, arr.Length - 1);  
  
            int i = ( arr.Length - 1 ) / 2;  
            int j = i + 1;  
  
            while ( ( i >= 0 ) &&   
                (j < arr.Length))  
            {  
                // 找到  
                if(arr[i] + arr[j]  == sum)  
                {  
                    Console.WriteLine(String.Format("{0} + {1} = {2}", arr[i], arr[j], sum));  
                    return true;  
                }  
                else if(arr[i] + arr[j] > sum)  
                {  
                    --i;  
                }  
                else // (arr[i] + arr[j]  < sum)  
                {  
                    ++j;  
                }  
            }  
  
            // goes here   
            Console.WriteLine("未能找到!");  
            return false;

 }

复制代码