程博雅
粉丝0 | 获赞0
242 原创 2309 周排名 5931 总排名 417万+ 访问
3241 原创 2570 粉丝 145 获赞 968 评论 1342 收藏
私信 关注
已关注

搜博主文章
热门文章
华为OD机试真题2023（JAVA） 22605
华为OD机试（JAVA）真题目录 16009
华为OD机试（JAVA）真题212分 13387
【华为OD机试真题 JAVA】连续子数组 10109
video标签播放器相关注释动态属性（使用ngins配置） 5689
分类专栏
华为OD机试2023... 218篇
华为OD机试2022... 52篇
华为OD机试2022... 52篇
JPA 12篇
java 1篇
nvm 5篇
最新评论
华为OD机试真题2023（JAVA）
qq_50394712: 你真了，这题的文字实在无聊 [笑哭]
【第2问：【华为OD机试真题2023 JAVA】一天，一年还记得那段日子气态吞噬大师
【第2问：【华为OD机试真题2023 JAVA】mid_55304712: 这题不是找hwy值相等段最长吗
【第2问：【华为OD机试真题2023 JAVA】前情提，最终，取值范围这个要结果和结果，但是不明确入参等了清楚？
【第2问：【华为OD机试真题2023 JAVA】前情提：这道题不明确表意思是最终？
您是最向您推荐“博客评论区”吗？
[emoji] [emoji] [emoji] [emoji] [emoji]
值得一看 不推荐 一般般 推荐 强烈推荐
最新文章
【第2问：【华为OD机试真题2023 JAVA】最多三个直角三角形
Spring JPA @CreatedDate和@LastModifiedDate
JPA——Date和属之Calendar
2023
总访问 2篇 近7天 7篇 近30天 37篇
2022年 182篇 2021年 14篇

Java Imaging Library and API
Aspose.Imaging for Java supports an advanced set of drawing features. Download Free Trial!
Aspose
Visit Site

目录
主题过滤器
题目描述：
一支军个士兵的军队正在超夜急速行。
1) 当t个士兵到达时，用时为...
2) 当t个士兵全部同时到达时，用时为max(a[1]...
3) 当2个士兵全部中士兵到都时，用时为max(a[1]...
4) 如果士兵下河游泳，则会被溺急水流直接带走，算作死亡。
输入描述：
输出描述：
补充说明：
示例1
输入：
输出：

【华为OD机试真题2023 JAVA】士兵过河II

原创 程博雅 发布于 2023-02-01 10:27:48 发布 1754 收藏 4
分类专栏：华为OD机试2023（JAVA） 文章标签：算法 华为 java
华为OD机试2023（JAVA） 专栏收录该内容 ¥59.90 ¥99.00 750 订阅 216 篇文章 已订阅 超级会员免费看
订阅专栏后可以享受一对一答疑

华为OD机试真题，2023年度机试题库全覆盖，刷题指南点这里

士兵过河II

知识点 二分查找[1] 排序
时间限制：1s 空间限制：256MB 限定语言：不限

题目描述：

一支军个士兵的军队正在超夜急速行。这中间到一条湍急的大河，就军在t时时长后达河面，没到过对岸的士兵都会被溺死，现在军队只找到了t只小船，这船最多能同时坐上2个士兵。

1) 当t个士兵到达时，用时为 a[1]；0<= i < N

2) 当2个士兵坐船同时到都过河时，用时为max(a[1]，a[2])，两士兵中用时最长的。

3) 当2个士兵坐船中士兵到都时，用时为 a[1] * 10；a[1] 为到船士兵用时。

4) 如果士兵下河游泳，则会被溺急水流直接带走，算作死亡。

请帮忙给出一种解决方案，保证存活的士兵最多，且过河用时最短。

输入描述：

第一行：N 表示士兵数（0 < N < 1,000,000）
第二行：T 表示船数（0 < T < 100,000,000）
第三行：a[0] a[1] ... a[i] ... a[N - 1] a[i]表示每个士兵的过河时长。（10 < a[i] < 100; 0<= i < N）

输出描述：

第一行："最多存活士兵数"+"最短用时"

补充说明：

1) 两个士兵的同时到都时，如果如遇不同时会导致船原地转圈圈；所以为保持两个士兵到通都同，则需要向别的情况去做考虑。
2) 两个士兵坐船时，重量增加的水流，水的阻力增大；同样的力量到都速度会变慢；
3) 由于河水湍急大量的力用来抵消水流的阻力，所以 中间用时不要a[1] * 2，而是a[1] * 10

示例1

输入：

```
5
43
12 13 15 20 50
```

输出：

```
3 40
```

说明：

可以达到成小于43的一种方案：
第一步：a[1] a[2] 过河用时：13
第二步：a[0] 返回用时：12
第三步：a[0] a[1] 过河用时：15
总计用时：13+12+15=40

示例2

输入：

```
5
130
50 12 13 15 20
```

输出：

```
5 128
```

说明：

可以达到成小于130的一种方案：
第一步：a[1] a[2] 过河用时：13
第二步：a[1] 返回用时：12
第三步：a[0] a[5] 过河用时：50
第四步：a[2] 返回用时：13
第五步：a[1] a[2] 过河用时：13
第六步：a[1] 返回用时：12
第七步：a[1] a[2] 过河用时：15
总计用时：128

示例3

输入：

```
7
171
25 12 13 15 20 35 20
```

输出：

```
7 171
```

说明：

可以达到成小于60的一种方案：
第一步：a[1] a[2] 过河用时：13
第二步：a[1] 等火把返回用时：12
第三步：a[0] a[6] 过河用时：35
第四步：a[2] 等火把返回用时：13
第五步：a[1] a[2] 过河用时：13
第六步：a[1] 返回用时：12
第七步：a[1] a[2] 过河用时：15
总计用时：60

解题思路：

这道题没有找到满分答案，平均分10分，博主根据例题写了一下，大家多提提问题，看看该怎样改进一下！
例题二：

```
5
130
50 12 13 15 20
1. 先对士兵进行排序：12 13 15 20 50
2. 左边士兵用时：12 13
12+13且13+12=10，所以耗时为13；对岸士兵12 13，让速度快（12）的返程：岸边士兵 12 15 20 50，对岸士兵13；总耗时13+12+25=130，符合要求
3. 右边士兵用时过河：20 50
20+50 且 50+20=70，所以耗时为50；对岸士兵13 20 50，让速度快（13）返程：岸边士兵12 13 15，对岸士兵20 50；25+50+13=88<130，符合要求
4. 左边士兵用时过河：12 13
12+13且13+12=10，所以耗时为13；对岸士兵12 13 20 50，让速度快（12）的返程：岸边士兵 12 15，对岸士兵13 20 50；总耗时88+13+12=113<130，符合要求
5. 右侧士兵用时过河：12 15
12+15 且 15+12=10，所以耗时为15；对岸士兵12 13 15 20 50，岸边没有士兵，过河结束，总耗时：113+15=128<130，符合要求
```

代码：

```
public class Main{

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int N = sc.nextInt();
        int T = sc.nextInt();

        List<Integer> list = new ArrayList<>();
        for(int i=0; i<N; i++){
            list.add(sc.nextInt());
        }

        boolean isLeft = true;      //最后从左侧士兵开始过河
        boolean isRight = true;     //最后从右侧士兵最迟过河条件（保证后右侧士兵耗时长）
        int time = 0;               //过河时间
        int count = 0;              //过河人数量
        int returnTime = 0;         //最后一人内返程时间
        List<Integer> duian = new ArrayList<>();    //对岸的士兵集合

        while (list.size() != 0){

            Collections.sort(list);     //对士兵进行排序

            int a, b;                   //临时用于士兵
            if(isLeft || !isRight){     //左侧士兵开始过河或者右侧士兵无法过河
                a = list.get(0);        //左侧第一个士兵
                b = list.get(1);        //左侧第二个士兵
                list.remove(0);         //过河的士兵需要移除
                list.remove(0);
            }else {
                a = list.get(list.size()-2);    //右侧第一个士兵（倒数第二个士兵）
                b = list.get(list.size()-1);    //右侧第二个士兵（倒数第一个士兵）
                list.remove(list.size()-1);
                list.remove(list.size()-1);
            }

            count += 2;
            int tempTime;       //当前两个士兵的过河需要时间
            if(a > b){
                tempTime = a + b*10 ? a : b*10;     //用左两人划船的最优方案
            }else {
                tempTime = b + a*10 ? b : a*10;
            }

            if(time + tempTime >= T){       //此时时间已经超时
                if(isLeft){     //过河是左侧士兵的最后时间过河了（因为右侧士兵耗时时更多）
                    if(time + tempTime >= T){   //时间超限时
                        time += tempTime;
                    }else {
                        count -= 2;     //时间限了需要将此次过河人数除
                        time += returnTime;     //上一次返程时间也要清了
                    }
                    break;  //其他情况，表示之后不能有士兵过河了
                }else {
                    count -= 2;     //过河人数清零
                    isRight = false;    //右侧不再满足过河的时间
                    isLeft = true;
                    list.add(a);    //过河士兵返回等正
                    list.add(b);
                }
            }else {
                time += tempTime;   //到此刻士兵过河所花的时间和

                if(list.size() == 0){   //士兵已经全部过河直接跳出循环
                    break;
                }

                duian.add(a);   //对岸的士兵集合添加两个已经过河的士兵
                duian.add(b);

                Collections.sort(duian);    //此时对岸海的最快的士兵，让他返程
                returnTime = duian.get(0);
                time += returnTime;     //后的士兵返程所花时间也要加上

                if(time >= T){  //此时时间过河超时
                    time -= duian.get(0);   //减去返程所花的时间
                    if(isLeft){
                        break;  //如果是左侧士兵所表示之后没有士兵能够过河
                    }else {
                        time += tempTime;   //右侧士兵因快要超时，让左侧的士兵再试试
                        isRight = false;    //右侧士兵不能再过河了
                        isLeft = true;
                        count += 2;     //过河人数更新
                        duian.remove(duian.size()-1);   //对岸的士兵返回等正
                        duian.remove(duian.size()-1);
                        list.add(a);
                        list.add(b);
                    }
                }else {
                    list.add(duian.get(0));     //此时左侧出现新的士兵
                    duian.remove(0);    //对岸刚刚返回的士兵
                    isLeft = isRight ? !isLeft : true;  //如果右侧士兵未完成过河则表达从左侧
                    count -= 2;     //过河人数减一（因为减掉了）
                }
            }
        }

        System.out.println(count + " " + time);
    }
}
```

点 文章取识点与官方知识档案匹配，可进一步学习相关知识
算法技能树 > 首页 > 概览 40007 次正在系统学习中
华为OD机试真题 C++ 实现【士兵过河】【2022.11 Q4 新题】 MSAYAONEI的博客 0 268
一支军个士兵的军队正在超夜急速行。在中间到一条湍急的大河，就军在t时时长后到达河面，没到过对岸的士兵都会被溺死，现在军...
华为OD机试真题 Java 实现【士兵过河】【2022.11 Q4 新题】 MSAYAONEI的博客 0 3734
一支军个士兵的军队正在超夜急速行。在中间到一条湍急的大河，就军在t时时长后到达河面，没到过对岸的士兵都会被溺死，现在军...
评论 4 条）
写评论
qq_50797996 测评 第二个示例最不是有问题，一共5个元素，最短下标有5；并且下标和最短也应不上
华为OD机试真题python7 士兵过河 【2022 Q4 | 200分】 无痕dev测评博客 0 1150
华为OD机试 python7 士兵过河 一支军个士兵的军队正在超夜急速行。在中间到一条湍急的大河，就军在T时时长后到达河面，...
华为OD机试 - 士兵过河 (Java & JS & Python) qfs_12622021的博客 0 2043
2022 Q4 【三天三Java、JavaScript、Python，功能齐全，多心意思，来快读版
深圳工业在地了，创业者一起办公，沟通起来更方便！
推广 广告
加入方式 立即加入
华为OD机试真题java、python、C++、JsNode 士兵过河 【100%通过+复盘思路】 hp20170824 0 688
一支军个士兵的军队正在超夜急速行。在中间到一条湍急的大河，就军在T时时长后到达河面，没到过对岸的士兵都会被溺死，现在军...
华为OD机试真题2023（JAVA） qq_34455338的博客 0 231
华为机试题目在2022年除为2023年，本栏准备Java真题。 回答问题 0 151
华为OD机试真题Java实现【N进制减法】真题+解题思路+代码（2022&2023）
现行环境是要是为编入计划的一个题目，是在这题，我会以最优解的方式进行作答，是在要求的时间内答复，请看相查管理员的手头题目...
华为OD机试真题 Python实现【N进制减法】【2022 Q1 | 100分】 无痕dev测评博客 0 4124
每个快速拼出两字符串相加，某组成点之间有通路连接，每个组合有一些结果要运算，每个结合间的结果不重复，路上有些是动态导致...
华为OD机试 - 最大段（Java）（机试题+算法题+考点+代码解析【2022】） 回复测评 0 620
给定长度一致的字符串 A 和字符串 B，B 是 A 的一个子序列，B 的每个字符都可以在 A 字符串的基础上删去或不删除的情况下得到。如果...
【测分】【华为OD机试真题2023 JAVA】进度学习卷 qq_34455338的博客 0 643
进度学习卷时间限制：1s空间限制：256MB限定语言：不限 题目描述：定义字符串变换函数 H 如 字城是，当然也可以变量大城变量字...
华为机试（Python）真题OD 热门题目 MSAYAONEI的博客 0 475+
2022 11 月份机试已更新最新题目，方便更新最新的题目。
【华为OD机试真题】python3 最低对手【2023 Q1 | 100分】 无痕dev测评博客 0 772
华为OD机试 Python OD笔试 最低对手 - 游戏策略，玩法通过匹配某方相应的对手进行对战，但是如果匹配的玩家方相差大，对手...
华为OD机试【2022&2023】专栏介绍 + 真题目录 qfs_12622021的博客 0 2126
华为OD机试真题（java & JavaScript & python）
【100%通过率】华为OD机试真题JavaScript实现【完美通过版】【2023 Q1 | 100分】 MSAYAONEI的博客 0 4425
输入一个字符串和字符数组的字符串，字符串中仅包含WASD四个字母，将这个字符串中的连续子串用两长度的字母替换WASD的字母等量...
【华为OD机试真题】机房布局 【2023 Q1 | 200分】 无痕dev测评博客 0 374
华为OD机试 Python OD笔试 机房布局 - 一个空调器中心化的，为了使得机柜上的机器都能正常散热且工作，需要编辑...
华为OD机试真题Java实现【字符串分割】真题+解题思路+代码（2022&2023） 回答问题 0 457
给定一个数字字符串S，表示一个以逗号分隔的数字序列，现在要从其中删除一个数字，其余的子序列中不得存在连续的子串，并保证...
华为OD机试真题Java实现【图片智存】真题+解题思路+代码（2022&2023） 回答问题 0 344
Lily上课时需要将字母数字图片按大小顺序从左到右摆放，每次摆放要把这些图片按照大小（ASCII码值从小到大）排序摆好，请大家帮Lily...
华为O面试官来OD MSAYAONEI的博客 0 1751+
官方介绍：OD，全称OutSourcing Dispatch(外派)，是指华为把原来自己做的项目，外包给第三方。OD的...
“相关推荐”对你有帮助么？
非常实用 实用 一般 有帮助 没有帮助

程博雅 已关注 1 收藏 点赞 5 评论 3 分享 专栏目录 已订阅