习题课1

邓丝雨



比赛题







第k小数

- · 给你一个长度为n的序列,求序列中第k小数的多少。
- t≤10,1≤n≤5×10^6

```
inline int read(){
  int x = 0, f = 1;
  char ch = getchar();
  while(ch < '0' || ch > '9'){
     if (ch == '-')
        f = -1;
     ch = getchar();
  while(ch >= '0' \&\& ch <= '9'){}
     x = (x << 1) + (x << 3) + (ch^48);
     ch = getchar();
  return x * f;
```



```
int finding(int 1, int r, int k)
 6 ⊟{
        if (1 == r) return a[1];
        int i = 1, j = r;
        int mid = (1 + r) / 2;
10
        int x = a[mid];
11
        while (i <= j)
12
13
            while (a[i] < x) i++;
14
            while (a[i] > x) i--;
15
             if (i <= j)
16
17
                 swap(a[i],a[j]);
18
                 i++; i--;
19
20
21
        if (k <= j) return finding(l, j, k);</pre>
22
        else if (i <= k) return finding(i, r, k);
23
        else return a[k];
24
```



● 不平行的直线

- ・在坐标纸上有N个不重合的点,两两可以连一个线段并延伸成直线,请问在这些直线里最多 能选出多少条使得他们两两不平行也不重合。
- N≤200,-1000≤Xi,Yi≤1000



基手绢

- ・牛客幼儿园的小朋友们围成了一个圆圈准备玩丢手绢的游戏,但是小朋友们太小了,不能围成一个均匀的圆圈,即每个小朋友的间隔可能会不一致。为了大家能够愉快的玩耍,我们需要知道离得最远的两个小朋友离得有多远(如果太远的话牛老师就要来帮忙调整队形啦!)。
- ・因为是玩丢手绢,所以小朋友只能沿着圆圈外围跑,所以我们定义两个小朋友的距离为沿着 圆圈顺时针走或者逆时针走的最近距离。
- 2≤N≤100000





一 二分

- ・我们刚刚学了二分查找——所谓二分查找就是在一堆有序数里找某个符合要求的数。在学完二分查找之后如果让你玩猜数游戏(裁判选定一个目标数字,你说一个数裁判告诉你是高了还是低了直到你猜到那个数)的话,显然你会用二分的方式去猜。
- 但是不是每一个玩猜数游戏的人都知道二分是最好,甚至一个健忘的玩家都有可能在得到裁判回答的下一个瞬间就忘了他之前问了什么以及裁判的回答),而现在更可怕的是,这个告诉你猜的数是高还是低的裁判他也很健忘,他总是薛定谔的记得这个目标数字,也就是说他的回答有可能出错。我们已经不关心这个不靠谱的游戏本身了,我们更关心裁判这个薛定谔的记得到底有几个是记得......
- ・现在给出这个健忘的玩家的所有猜测和裁判的所有回答,问裁判最多能有多少次是记得目标 数字的,即裁判的回复是符合情况的。
- n≤100000



● 还是猜数字:

- ・ 每次猜两个数字X1,X2 (保证两个数字不同奇偶性) 然后,裁判告诉你哪个数字更接近目标数字。
- · 猜了N次,且根据这N次的结果已经能判断出这个数字是多少了,但是玩家并没有猜出来。
- ・ 现在把他的N次猜测给你,但是不告诉你裁判的反应,问这个数可能有哪些。
- ・Eg: 在1-4当中猜,猜三次分别是 (1, 2) (2, 3) (1, 4) 答案是2
- ・如果目标数字是1: (1, 2) 回答1 (已经确认答案)
- 目标数字是2: (1, 2) 回答2, (2, 3) 回答2 (确定答案)
- ・目标数字是3:(1,2)回答2,(2,3)回答3,(1,4)回答4——还不能确定答案
- ・目标数字是4: (1, 2) 回答2, (2, 3)回答3, (1, 4) 回答4——不能确定



● 交换

- ・牛客幼儿园的小朋友课间操时间需要按照学号从小到大排队,但是他们太小了只能站成一列 顺序却不对,现在幼儿园的阿姨需要帮忙交换小朋友的位置让他们最终有序,阿姨希望能尽 快完成交换操作,问最少需要交换多少次,才能使得小朋友们从小到大排好。
- · 注意: 每个小朋友的学号不同, 但是未必连续, 因为可能有小朋友请假了没有来。
- N≤100000

- · 132546
- · 134526



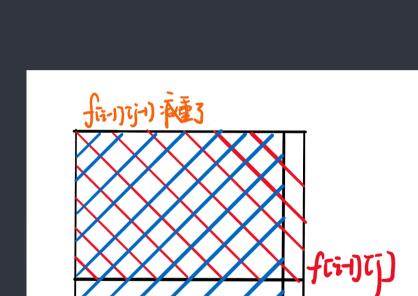
课后题



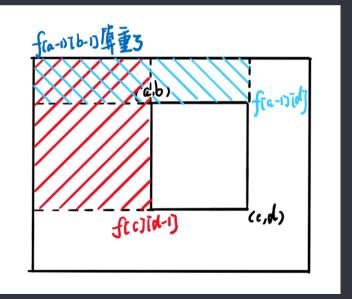
[HNOI2003]激光炸弹

- ・一种新型的激光炸弹,可以摧毁一个边长为R的正方形内的所有的目标。
- ・现在地图上有 $n(N \le 10000)$ 个目标,用整数 X_i, Y_i (其值在[0,5000])表示目标在地图上的位置,每个目标都有一个价值。
- ・激光炸弹的投放是通过卫星定位的,但其有一个缺点,就是其爆破范围,即那个边长为R的正 方形的边必须和x,y轴平行。
- ・若目标位于爆破正方形的边上,该目标将不会被摧毁。





रिकार्।-1)





NC50999 表达式计算4

- ・ 给出一个表达式,其中运算符仅包含+,-,*,/,^ (加 减 乘 整除 乘方) 要求求出表达式的最终值
- · 数据可能会出现括号情况,还有可能出现多余括号情况
- ・数据保证不会出现≥2^31的答案
- 数据可能会出现负数情况



NC19916 [CQOI2010]扑克牌

・你有n种牌,第i种牌的数目为ci。另外有一种特殊的牌: joker, 它的数目是m。你可以用每种牌各一张来组成一套牌, 也可以用一张joker和除了某一种牌以外的其他牌各一张组成1套牌。比如, 当n=3 时, 一共有4种合法的套牌: {1,2,3}, {J,2,3}, {1,J,3}, {1,2,J}。 给出n, m和ci, 你的任务是组成尽量多的套牌。每张牌最多只能用在一副套牌里(可以有牌不使用)。



● NC19913 中位数图

・ 给出1~n的一个排列,统计该排列有多少个长度为奇数的连续子序列的中位数是b。中位数是 指把所有元素从小到大排列后,位于中间的数。



b如果要是中位数的话,大于他的数和小于他的数个数就需要一样,因为已经知道b,那么我们可以把大于b的数记为1,小于b的数记为-1,等于他的数记为0,那么我们现在求的就是区间和为0且里面有0的长度为奇数的区间个数(因为1和-1个数要一样,且还有一个0,所以长度为奇数这个条件在前两个条件里已经满足了)。

我们可以以0作为起点(只有一个0)分别看他的左边和右边,统计从他开始的左边后缀和为i的个数,和右他开始的边前缀和为i的个数,显然当左边后缀和为x的时候,右边的前缀和要为-x才能保证区间和为0,统计即可。



NC14301 K-th Number

· 对数列A的每个区间求第K大,并将第k大插入到B中,再求B的第M大。



・把求值变成验证——

首先二分答案,每次二分的时候我们得到一个x,这个时候我们利用尺取,去得到第K大数大于x的区间一共有多少,如果大于m-1个说明x取小了。

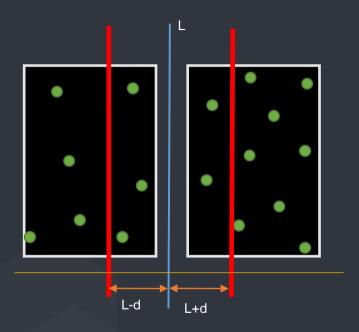
那么第k大数大于x的区间一共有多少怎么求? 第k大数大于x的其实也就是大于x的数至少有k个,当我们枚举区间左界L,我们可以从L往右扫描到第一个大于x的数有k个的的点R,右界在这个位置及其之后的区间大于x的点的个数都大于等于k个。而当L右移一个位置,R显然只会右移不会左移,所以L和R对每一个数字都只访问一遍,这样,时间复杂度是 O(n) 的了。加上二分就是 O (nlogn) 了。

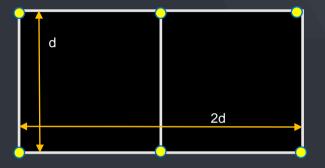


最近点对Raid

· 在二维平面上的n个点中, 找出距离最近的两个点。









```
31
      double near dis(int 1, int r)
32
33
          if(r == l+1) return dis(p[l], p[r]);
34
          if(1+2 == r) return min(dis(p[1], p[r]), min(dis(p[1], p[1+1]), dis(p[1+1], p[r])));
35
          int mid = (1 + r) \gg 1;
36
          double ans = min(near dis(l, mid), near dis(mid+1, r));
37
          int cnt = 0;
38
          for(int i = 1; i \le r; i++)
39
              if(p[i].x >= p[mid].x - ans && p[i].x <= p[mid].x + ans)
40
                  a[cnt++] = i;
41
          sort(a, a+cnt, compy);
42
          for(int i = 0; i < cnt; i++)
43
          for (int j = i + 1; j < cnt; j++)
44
45
              if(p[a[j]].y - p[a[i]].y >= ans) break;
46
              ans = min(ans, dis(p[a[i]], p[a[j]]));
47
48
          return ans;
49
```



- ・第一节 https://ac.nowcoder.com/discuss/428998
- 第二节 https://ac.nowcoder.com/discuss/429002
- 第三节 https://ac.nowcoder.com/discuss/430437
- ・比赛 https://ac.nowcoder.com/discuss/431780



Thanks

