

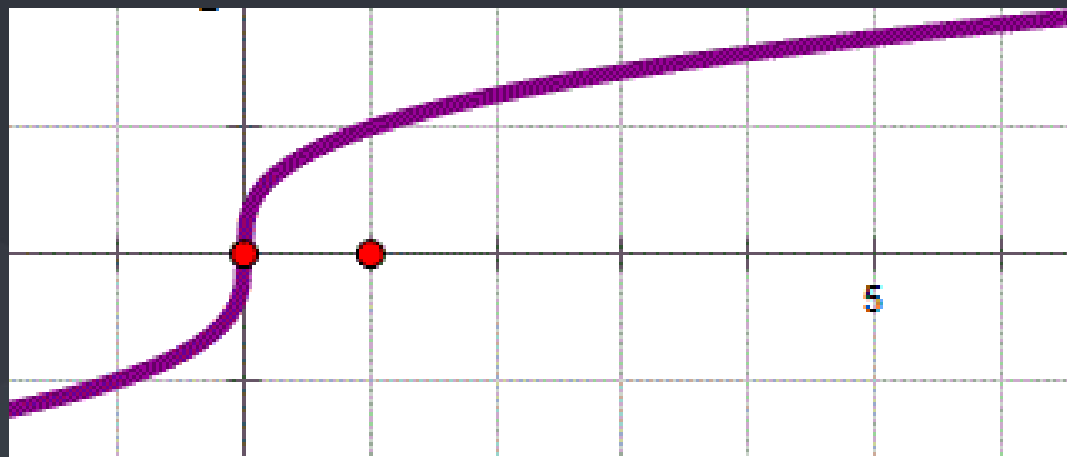
算法基础——二分三分01分数规划

邓丝雨



二分

- 单调函数求0点





二分查找

- 在一个**单调有序**的集合中查找元素，每次将**集合**分为左右两部分，判断解在哪个部分中并调整**集合**上下界，重复直到找到目标元素。
- 例如在以下序列中查找55

0	5	13	19	22	41	55	68	72	81	98
---	---	----	----	----	----	----	----	----	----	----

55	68	72	81	98
----	----	----	----	----

55	68
----	----

55





C++ STL的二分查找函数

- `binary_search` 返回bool值,是否存在
 - `lower_bound` 返回可插入的最小位置的迭代器
 - 即返回第一个符合条件的元素位置
 - `upper_bound` 返回可插入的最大位置的迭代器
 - 即返回最后一个符合条件的元素位置
-
- Eg:
 - `lower_bound(a, a+11, 55)`





例1:

- 给一串 n 个单调递增的数，有 q 次询问 $\geq x$ 且 $\leq y$ 的数有多少个
- 数据规模: $1 \leq n \leq 10^5$ $1 \leq q \leq 50000$



```
14  int find_low(int x)
15  {
16      int mid, l = 1, r = n;
17      while (l <= r)
18      {
19          mid = (l + r) >> 1;
20          if (a[mid] < x) l = mid + 1;
21          else r = mid - 1;
22      }
23      return l;
24  }
25  int find_up(int x)
26  {
27      int mid, l = 1, r = n;
28      while (l <= r)
29      {
30          mid = (l + r) >> 1;
31          if (a[mid] <= x) l = mid + 1;
32          else r = mid - 1;
33      }
34      return l;
35  }
```



例2:

- 给你N个机器和M个任务，每个任务有两个值花费时间x和难度y，每个机器也有两个值最大工作时间x1和最大工作难度y1，机器可以胜任某个工作的条件是 $x1 \geq x \ \&\& \ y1 \geq y$ ，机器胜任一个工作可以拿到 $x*500+2*y$ 的钱，现在问你怎么匹配才能使匹配数最大且钱数最多。
- $X_i \leq 1440 \ y_i \leq 100$
- $1 \leq N \leq 100000, 1 \leq M \leq 100000$





二分答案+检验

- 对于难以直接确定解的问题,采取**二分枚举+检验**的思想将求解类问题转换为验证类问题
- $\geq K$ 可行
- $< K$ 不可行
- 目标就是求K





例3:

- 有N个牛棚在x轴上,已知他们的坐标.FJ有C只奶牛,每只都必须安排在一个牛棚里,一个牛棚只能容纳一只.但是他们会互相攻击,所以要求距离最近的两个牛棚间的距离最大.
- $2 \leq N \leq 100,000$
- $0 \leq x_i \leq 1,000,000,000$
- $2 \leq C \leq N$





例3:

- 我们可以假设距离最近的两个牛棚间的距离为 x , 你能判断这个 x 是否可行吗?
- 先把 x_i 排序
- 对于一个解 m , 我们验证 m 是否可行.
- 第一个点放置牛.
- 从左向右 $O(N)$ 扫描一遍牛棚, 第 i 个点如果和上一个放置点的距离 $\geq m$ 则在第 i 个点放置一头牛, 统计总放置的数量是否 $\geq C$.





24
25
26
27
28
29
30
31
32
33
34
35
36
37
38

```
while (scanf("%d%d", &n, &c) != EOF)
{
    for (int i = 1; i <= n; i++) scanf("%d", &a[i]);
    sort(a + 1, a + n + 1);

    int mid, l = 1, r = a[n];
    while (l <= r)
    {
        mid = (l + r) >> 1;
        if (judge(mid)) l = mid + 1;
        else r = mid - 1;
    }
    cout << r << endl;
}
return 0;
```

```
7 bool pan(int x)
8 {
9     int la = a[1];
10    int cnt = 1;
11    for (int i = 2; i <= n; i++)
12    {
13        if (a[i] - la >= x)
14        {
15            la = a[i];
16            cnt++;
17        }
18        if (cnt >= c) return true;
19    }
20    return false;
21 }
```



牛客竞赛

AC.NOWCODER.COM



例4:

- 有 n 件衣服需要晾干,每件含水量 a_i .每件衣服每分钟自然干1单位的水.每分钟可以对其中任意一件使用吹风机,其可以减少 k 的水.求晾干所有衣服的最少时间.
- $1 \leq n \leq 100\ 000$
- $1 \leq a_i \leq 10^9$
- $1 \leq k \leq 10^9$





- 设某次二分出的一个值是mid:
- 1、对于一件ai值小于等于mid的衣服, 直接晾干即可;
- 2、对于一件ai值大于mid值的衣服, 最少的用时是用吹风机一段时间, 晾干一段时间, 设这两段时间分别是x1和x2, 那么有 $mid = x1 + x2$, $ai \leq k * x1 + x2$, 解得 $x1 \geq (ai - mid) / (k - 1)$, 所以对 $(ai - mid) / (k - 1)$ 向上取整就是该件衣服的最少用时。
- 我们把所有的使用吹风机时间加起来, 这个总和应该 $\leq mid$





- 解决最大值最小 or 最小值最大的常见方法





例5:

- 题意：给你N行4列的数，从每一列选一个数，问使他们的和为0的情况有多少种
($N \leq 4000$)



- 既然有四列，那么我们可以分别计算前两列和后两列的和（只需要 $n*n*2$ 次运算），然后对后两列的和排序，那么我们对于每一个前两列的和都可以二分找到后两列的和中与之相加为0的个数，这样的复杂度就是 $O(n*n*\log(n))$ 是可以接受的，





三分

类似二分的定义Left和Right

$\text{mid} = (\text{Left} + \text{Right}) / 2$

$\text{midmid} = (\text{mid} + \text{Right}) / 2$;

如果mid靠近极值点, 则 $\text{Right} = \text{midmid}$;

否则(即midmid靠近极值点), 则 $\text{Left} = \text{mid}$;



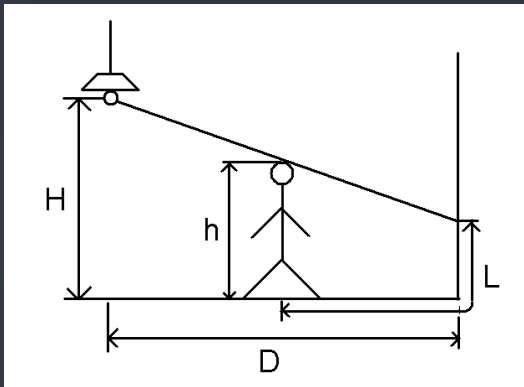
牛客竞赛

AC.NOWCODER.COM



例1:

- 如图，人左右走动，求影子 L 的最长长度。
- 根据图，很容易发现当灯，人的头部和墙角成一条直线时(假设此时人站在A点)，此时的长度是影子全在地上的最长长度。当人再向右走时，影子开始投影到墙上，当人贴着墙，影子长度即为人的高度。所以当人从A点走到墙，函数是先递增再递减，为凸性函数，所以我们可以用三分法来求解。





例2:

- 在一个2维平面上有两条传送带，每一条传送带可以看成是一条线段。两条传送带分别为线段AB和线段CD。lxhgww在AB上的移动速度为P，在CD上的移动速度为Q，在平面上的移动速度R。现在lxhgww想从A点走到D点，他想知道最少需要走多长时间。





01分数规划

- 有一堆物品，每一个物品有一个收益 a_i ，一个代价 b_i ，我们要求一个方案选若干个物品使得所选择的 $\frac{\sum a_i}{\sum b_i}$ 最大。



Thanks

