



数据结构和算法 (Python描述)

郭 炜

微信公众号



微博: <http://weibo.com/guoweiofpku>

学会程序和算法，走遍天下都不怕!

讲义照片均为郭炜拍摄



贪心算法



北京大学
PEKING UNIVERSITY

信息科学技术学院

例题:

Santa Clau' s Gifts



奥地利萨尔茨堡

圣诞老人的礼物-Santa Clau' s Gifts(百练4110)

圣诞节来临了，圣诞老人准备分发糖果，现在有多箱不同的糖果，每箱糖果有自己的价值和重量，每箱糖果都可以拆分成任意散装组合带走。圣诞老人的驯鹿雪橇最多只能装下重量 W 的糖果，请问圣诞老人最多能带走多大价值的糖果。



输入

第一行由两个部分组成，分别为糖果箱数正整数 n ($1 \leq n \leq 100$)，驯鹿能承受的最大重量正整数 w ($0 < w < 10000$)，两个数用空格隔开。其余 n 行每行对应一箱糖果，由两部分组成，分别为一箱糖果的价值正整数 v 和重量正整数 w ，中间用空格隔开。

输出

输出圣诞老人能带走的糖果的最大总价值，保留1位小数。输出为一行，以换行符结束。



样例输入

4 15

100 4

412 8

266 7

591 2

样例输出

1193.0



圣诞老人的礼物-Santa Claus Gifts(百练4110)

解法：

按礼物的价值/重量比从大到小依次选取礼物，对选取的礼物尽可能多地装，直到达到总重量 w

复杂度： $O(n\log n)$

圣诞老人的礼物-Santa Clau' s Gifts(百练4110)

```
eps = 1e-6
class Candy:
    def __init__(self, v=0, w=0 ):
        self.v = v
        self.w = w
    def __lt__(self, other):
        return (self.v / self.w - other.v / other.w) > eps

n, w = list( map( int, input().split() ) )
candies = [ Candy() for i in range(n) ]
for i in range(n):
    candies[i].v, candies[i].w = list( map( float, input().split() ) )
candies.sort()
totalW = 0
totalV = 0
```


圣诞老人的礼物-Santa Claus Gifts(百练4110)

```
for i in range(n):
    if (totalW + candies[i].w) <= w:
        totalW += candies[i].w
        totalV += candies[i].v
    else:
        totalV += candies[i].v * float(w-totalW)/candies[i].w
        break
print('%.1f'%totalV)
```

圣诞老人的礼物-Santa Claus's Gifts(百练4110)

证明:

替换法。对于用非此法选取的最大价值糖果箱序列，
可以将其按价值/重量比从大到小排序后得到：

序列1: a_1, a_2, \dots

用序列1和按上述解法选取的序列2依次进行比较：

序列2: b_1, b_2, \dots

价值/重量比相同的若干箱糖果，可以合并成一箱，所以两个序列中元素都不重复

对于发现的第一个 $a_i \neq b_i$ ，则必有： $a_i < b_i$

则在序列1中，用 b_i 这种糖果，替代若干重量的 a_i 这种糖果，则会使得序列1的总价值增加，这和序列1是价值最大的取法矛盾

所以：序列1 = 序列2 （序列2不可能是序列1的一个前缀且比序列1短）

贪心算法

每一步行动总是按某种指标选取最优的操作来进行，该指标只看眼前，并不考虑以后可能造成的影响。

贪心算法需要证明其正确性。

“圣诞老人礼物”题，若糖果只能整箱拿，则贪心法错误。

贪心算法

每一步行动总是按某种指标选取最优的操作来进行，该指标只看眼前，并不考虑以后可能造成的影响。

贪心算法需要证明其正确性。

“圣诞老人礼物”题，若糖果只能整箱拿，则贪心法错误。

考虑下面例子：

3个箱子(8,6) (5,5) (5,5)，雪橇总容量10



北京大学
PEKING UNIVERSITY

信息科学技术学院

北京大学信息学院 郭炜

例题:电影节



维也纳美泉宫

例题：电影节 (百练4151)

大学生电影节在北大举办! 这天, 在北大各地放了多部电影, 给定每部电影的放映时间区间, 区间重叠的电影不可能同时看 (端点可以重合), 问李雷最多可以看多少部电影。



例题：电影节(百练4151)

输入

多组数据。每组数据开头是 n ($n \leq 100$)，表示共 n 场电影。
接下来 n 行，每行两个整数(均小于1000)，表示一场电影的放映区间
 $n=0$ 则数据结束

输出

对每组数据输出最多能看几部电影



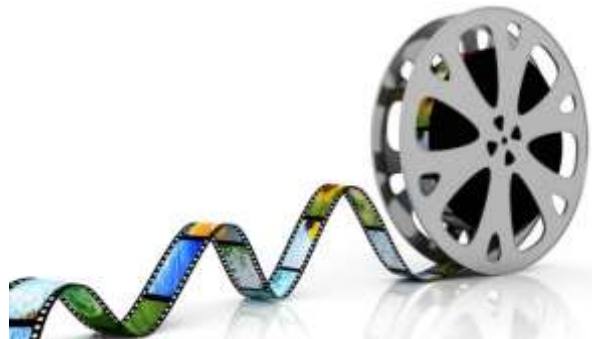
例题：电影节(百练4151)

Sample Input

```
12
1 3
3 4
0 7
3 8
15 19
15 20
10 15
8 18
6 12
5 10
4 14
2 9
0
```

Sample Output

```
5
```



例题：电影节(百练4151)

贪心解法

将所有电影按结束时间从小到大排序，第一步选结束时间最早的那部电影。然后，每步都选和上一部选中的电影不冲突且结束时间最早的电影。

复杂度： $O(n \log n)$



例题：电影节(百练4151)

证明：

替换法。假设用贪心法挑选的电影序列为：

a_1, a_2, \dots

不用此法挑选的最长的电影序列为：

b_1, b_2, \dots

现可证明，对任意 i ， b_i 均可以替换成 a_i

例题：电影节(百练4151)

用 $S(x)$ 表示 x 开始时间， $E(x)$ 表示 x 结束时间，则：

- 1) b_1 可以替换成 a_1 ，因为 $E(a_1) \leq E(b_1)$
- 2) 若可以找到 a_i ，满足 $E(a_i) \leq E(b_i)$ 且 a_i 可以替换 b_i ，则 $E(a_{i+1}) \leq E(b_{i+1})$ 且 a_{i+1} 可以替换 b_{i+1}

证：

因为 $E(a_i) \leq E(b_i)$ 且 $E(b_i) \leq S(b_{i+1})$

则： $E(a_i) \leq S(b_{i+1})$

a_{i+1} 是所有 $S(x) \geq E(a_i)$ 的 x 中， $E(x)$ 最小的

$S(b_{i+1}) \geq E(b_i) \geq E(a_i)$ ，所以 $E(b_{i+1}) \geq E(a_{i+1})$

因此用 a_{i+1} 替换 b_{i+1} 不会对后续造成影响，替换可行



北京大学
PEKING UNIVERSITY

信息科学技术学院

北京大学信息学院 郭炜

例题 Radar Installation



瑞士卢塞恩

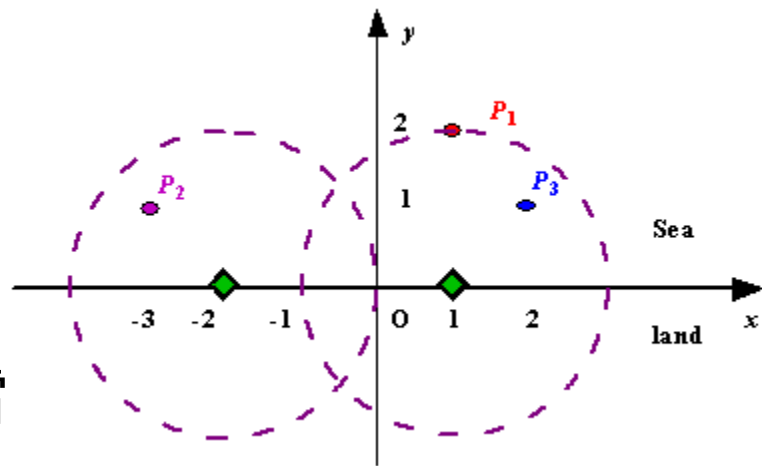
例题: Radar Installation(百练1328)

x 轴是海岸线, x 轴上方是海洋。海洋中有 n ($1 \leq n \leq 1000$) 个岛屿, 可以看作点。

给定每个岛屿的坐标 (x, y) , x, y 都是整数。

当一个雷达 (可以看作点) 到岛屿的距离不超过 d (整数), 则认为该雷达覆盖了该岛屿。

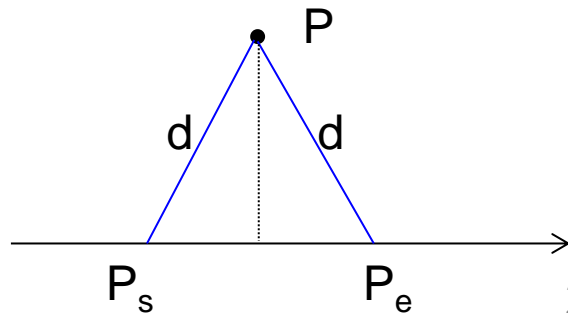
雷达只能放在 x 轴上。问至少需要多少个雷达才可以覆盖全部岛屿。



例题: Radar Installation(百练1328)

对每个岛屿 P ,可以算出, 覆盖它的雷达, 必须位于 x 轴上的区间 $[P_s, P_e]$ 。

如果有雷达位于某个 x 轴区间 $[a, b]$, 称该雷达覆盖此区间。问题转换为, 至少要在 x 轴上放几个雷达(点), 才能覆盖全部区间 $[P1_s, P1_e], [P2_s, P2_e] \dots [Pn_s, Pn_e]$



例题: Radar Installation(百练1328)

重要结论:

如果可以找到一个雷达同时覆盖多个区间, 那么把这多个区间按起点坐标从小到大排序, 则雷达放在最后一个区间(起点最靠右的) k 的起点, 就能覆盖所有区间

证明: 如果它不能覆盖某个区间 x , 那么它必然位于 1) x 起点的左边, 或者 2) x 终点的右边。

情况1) 和 k 的起点是最靠右的矛盾

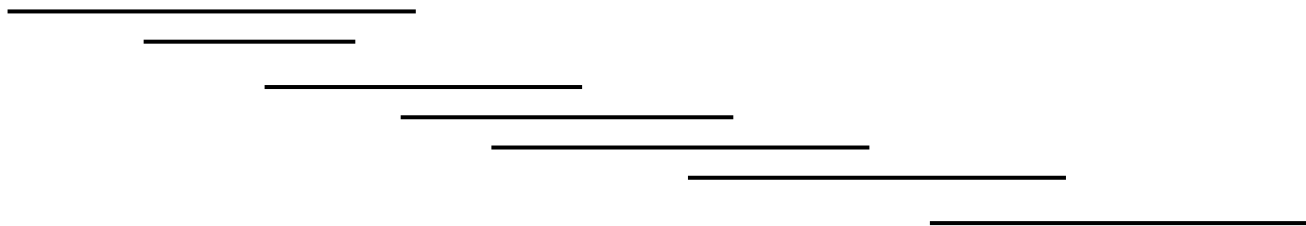
情况2) 如果发生, 则不可能找到一个点同时覆盖 x 和 k , 也和前提矛盾

有了这个结论, 就可以只挑区间的起点来放置雷达了。

例题: Radar Installation(百练1328)

贪心算法:

- 1) 将所有区间按照起点从小到大排序, 并编号 $\text{segs}[0] - \text{segs}[n-1]$
- 2) 依次考察每个区间的起点, 看要不要在那里放雷达。开始, 所有区间都没被覆盖, 设置目前未被覆盖的区间中右端点的最小坐标 noCoveredMinX2 为0号区间右端点坐标 $\text{segs}[0].x2$



例题: Radar Installation(百练1328)

3) 考察一个区间 $\text{segs}[i]$ 的起点 $\text{segs}[i].x1$ 的时候, 要看 noCoveredMinX2 是否小于 $\text{segs}[i].x1$ 。如果不是, 则先不急于在 $\text{segs}[i]$ 放雷达, 设置

$$\text{noCoveredMinX2} = \min(\text{noCoveredMinX2}, \text{segs}[i].x2)$$

接着往下看.

如果是, 那么就必须往 $\text{segs}[i-1]$ 放一个雷达了 (往后面的区间放雷达, 都不可能覆盖 noCoveredMinX2 所在区间了)。放好后, $\text{segs}[i]$ 成为第一个可能未被覆盖的区间, 因此设置

$$\text{noCoveredMinX2} = \text{segs}[i].x2$$

然后再继续。最后一个区间上要放一个雷达。

例题： Radar Installation(百练1328)

证明：

替换法。考虑不用贪心法获得的最佳雷达摆放方案。将其所有雷达按坐标从小到大排序得到 x_1, x_2, \dots 。用贪心法得到的雷达坐标从小到大排序则为 y_1, y_2, \dots 。

可证明每个 x_i 都可以被 y_i 替换，且 y 序列不会比 x 序列长。

先证明 x_1 可以用 y_1 替换。

用 $S(x)$ 表示贪心法中区间 x 的起点, 假设 $y_1 = S(i)$

a) 若 $x_1 < y_1$, 则用 y_1 替换 x_1 没问题, 因 y_1 覆盖了区间0到 i , x_1 覆盖的区间更少

b) 若 $x_1 > y_1$, 则分两种情况讨论:

1) $x_1 < S(i+1)$: 因 x_1 不能覆盖 $i+1$ 及以后区间, 且 i 及以前的区间已经被 y_1 覆盖, 所以将 x_1 用 y_1 替换, 不会有损失。

2) $x_1 \geq S(i+1)$

贪心法中, 在区间 i 起点放雷达, 是因为如果不放而在 $S(i+1)$ 放雷达, 则该雷达不能覆盖 i 及其前面的某个区间 C 。

若 $x_1 \geq S(i+1)$, 则 x_1 也不能覆盖 C 。 $x_i (i > 1)$ 更加不能。因此 $x_1 \geq S(i+1)$ 是不可能的。

类似地证明, 假设 x_i 可以被 y_i 替换, 则 x_{i+1} 也可以被 y_{i+1} 替换。