# Assignment #5: Greedy 穷举 Implementation

Updated 1939 GMT+8 Oct 21, 2024

2024 fall, Complied by <mark>同学的姓名、院系</mark>

\*\*说明：\*\*

1）请把每个题目解题思路（可选），源码 Python, 或者 C++（已经在 Codeforces/Openjudge 上 AC），截

图（包含 Accepted ），填写到下面作业模版中（推荐使用 typora https://typoraio.cn ，或者用

word）。AC 或者没有 AC，都请标上每个题目大致花费时间。

3）提交时候先提交 pdf 文件，再把 md 或者 doc “ ”文件上传到右侧 作业评论 。Canvas 需要有同学清晰头像、提

交文件有 pdf、"作业评论"区有上传的 md 或者 doc 附件。

4）如果不能在截止前提交作业，请写明原因。

## 1. 题目

### 04148: 生理周期

brute force, http://cs101.openjudge.cn/practice/04148

思路：p e i不相等的时候就让它们玩命加一直加到相等，还好这题数不大，顶多加个几百次，所以不会超时

代码：

```python

k=0

while True:

k+=1

p,e,i,d=map(int,input().split())

if p==e==i==d==-1:

break

elif p==e==i:

result=21252-d

else:

while True:

if p<e:

p+=23

if e<p:

e+=28

if p<i:

p+=23

if i<p:

i+=33

if e<i:

e+=28

if i<e:

i+=33

if p==e==i:

break

result=p-d

print(f'Case {k}: the next triple peak occurs in {result} days.')

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 18211: 军备竞赛

greedy, two pointers, http://cs101.openjudge.cn/practice/18211

思路：把便宜的留在自己这边，把贵的卖给对面赚更多money，用双指针来弄还算比较简单的，不过要注意只剩最后一件没钱做的时候别卖给对面（也就是代码中的i!=j）

代码：

```python

p=int(input())

cost=list(map(int,input().split()))

cost.sort()

i=0

j=len(cost)-1

while i<=j:

if p>=cost[i]:

p-=cost[i]

i+=1

elif len(cost)-1-j<i and i!=j:

p+=cost[j]

j-=1

else:

break

print(i-len(cost)+1+j)

```

代码运行截图 ==（至少包含有"Accepted"）==

### 21554: 排队做实验

greedy, http://cs101.openjudge.cn/practice/21554

思路：让耗时短的同学先做，总等待时间就是第一个人做实验时间\*（n-1）+第二个人\*（n-2）+......，最后求平均除以n

代码：

```python

n=int(input())

time=list(map(int,input().split()))

queue=sorted(enumerate(time,1),key=lambda x:x[1])

print(\*[x[0] for x in queue])

t=0

for i in range(n-1):

t+=queue[i][1]\*(n-i-1)

print('%.2f'%(t/n))

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 01008: Maya Calendar

implementation, http://cs101.openjudge.cn/practice/01008/

思路：感觉很烦人的一道题目，光是读题就读了半天，还有个烦人的地是有的地方从0开始有的从1开始，大致的思路就是先把总共的天数算出来再跟另一种纪年法对应

代码：

```python

months=['pop', 'no', 'zip', 'zotz', 'tzec', 'xul', 'yoxkin', 'mol', 'chen', 'yax', 'zac', 'ceh', 'mac', 'kankin', 'muan', 'pax', 'koyab', 'cumhu']

months1=['imix', 'ik', 'akbal', 'kan', 'chicchan', 'cimi', 'manik', 'lamat', 'muluk', 'ok', 'chuen', 'eb', 'ben', 'ix', 'mem', 'cib', 'caban', 'eznab', 'canac','ahau']

day=0

n=int(input())

print(n)

for i in range(n):

a,b,c=map(str,input().split())

a=list(a)

a.remove('.')

a=int(''.join(a))

if b in months:

day+=months.index(b)\*20

else:

day+=360

day+=a

day+=int(c)\*365

year=day//260

dayinyear=day%260

month=months1[dayinyear%20]

date=dayinyear%13+1

print(f'{date} {month} {year}')

day=0

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 545C. Woodcutters

dp, greedy, 1500, https://codeforces.com/problemset/problem/545/C

思路：第一棵和最后一棵都能砍（不过还要注意只有一棵树的情况。。。），中间的树尽量往左边倒（因为这样不会影响后面的树是否能被砍）

代码：

```python

n=int(input())

trees=[]

trees1=[]

cnt=2

for i in range(n):

x,h=map(int,input().split())

trees.append((x,h))

trees1.append(x)

if n<3:

print(n)

else:

for i in range(1,n-1):

if trees[i][0]-trees[i][1]>trees1[i-1]:

cnt+=1

elif trees[i][0]+trees[i][1]<trees1[i+1]:

trees1[i]+=trees[i][1]

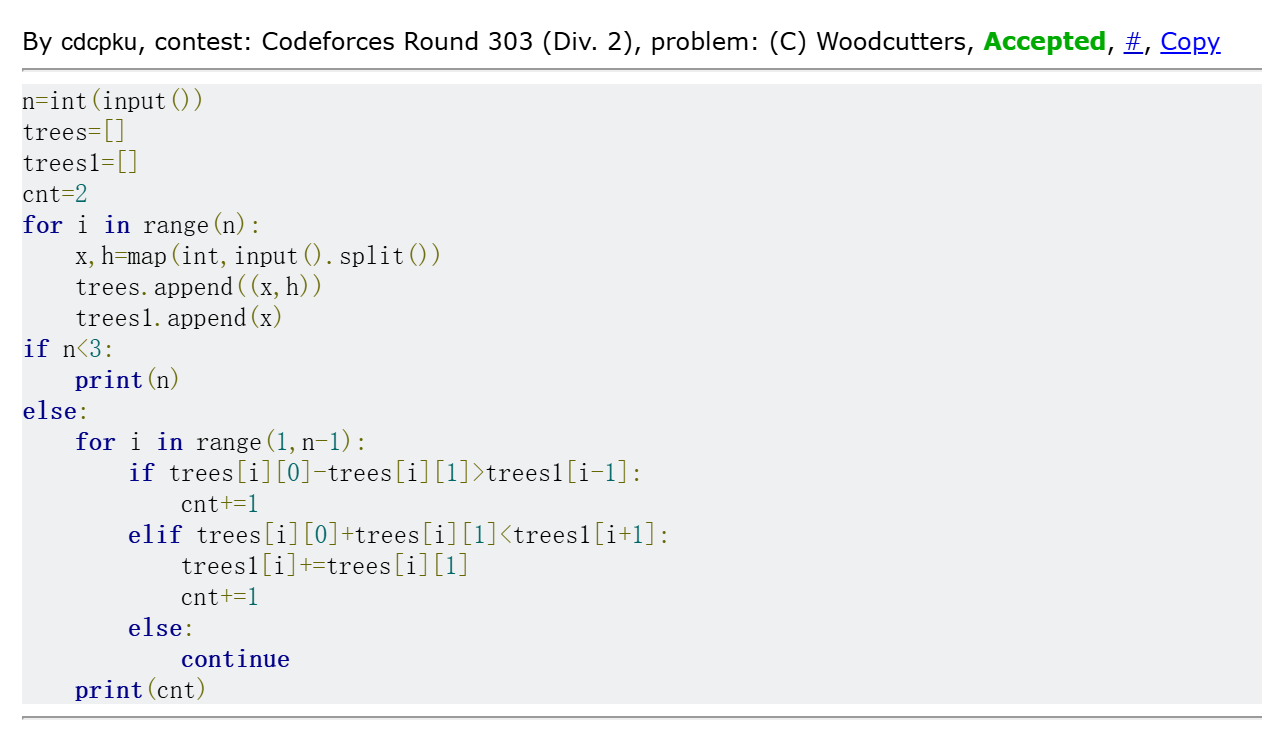
cnt+=1

else:

continue

print(cnt)

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

### 01328: Radar Installation

greedy, http://cs101.openjudge.cn/practice/01328/

思路：对于每座岛，可以以岛为圆心画一个半径为d的圆，与x轴交于两点（也可能是一个），雷达只要在这两点之间即可，把每座岛对应的区间都找到，如果雷达位于多个区间的交集，就可以满足多个岛的需求，要做的就是找这个交集数量的最小值（一开始rte，看了眼群里别人发的代码才发现两组数据之间的空行也得输入进去，接着又presentation error，原来是输出的冒号后面还有个空格，感觉这题坑还挺多）

代码：

```python

from math import sqrt

a=0

while True:

a+=1

n,d=map(int,input().split())

if n==d==0:

break

else:

X=[]

Y=[]

for i in range(n):

x,y=map(int,input().split())

X.append(x)

Y.append(y)

useless=input()

if max(Y)>d:

print(f'Case {a}: -1')

else:

radar=[]

cnt=1

for i in range(n):

radar.append([X[i]-sqrt(d\*\*2-Y[i]\*\*2),X[i]+sqrt(d\*\*2-Y[i]\*\*2)])

radar=sorted(radar,key=lambda x:x[1])

right=radar[0][1]

for i in range(1,n):

if radar[i][0]<=right:

continue

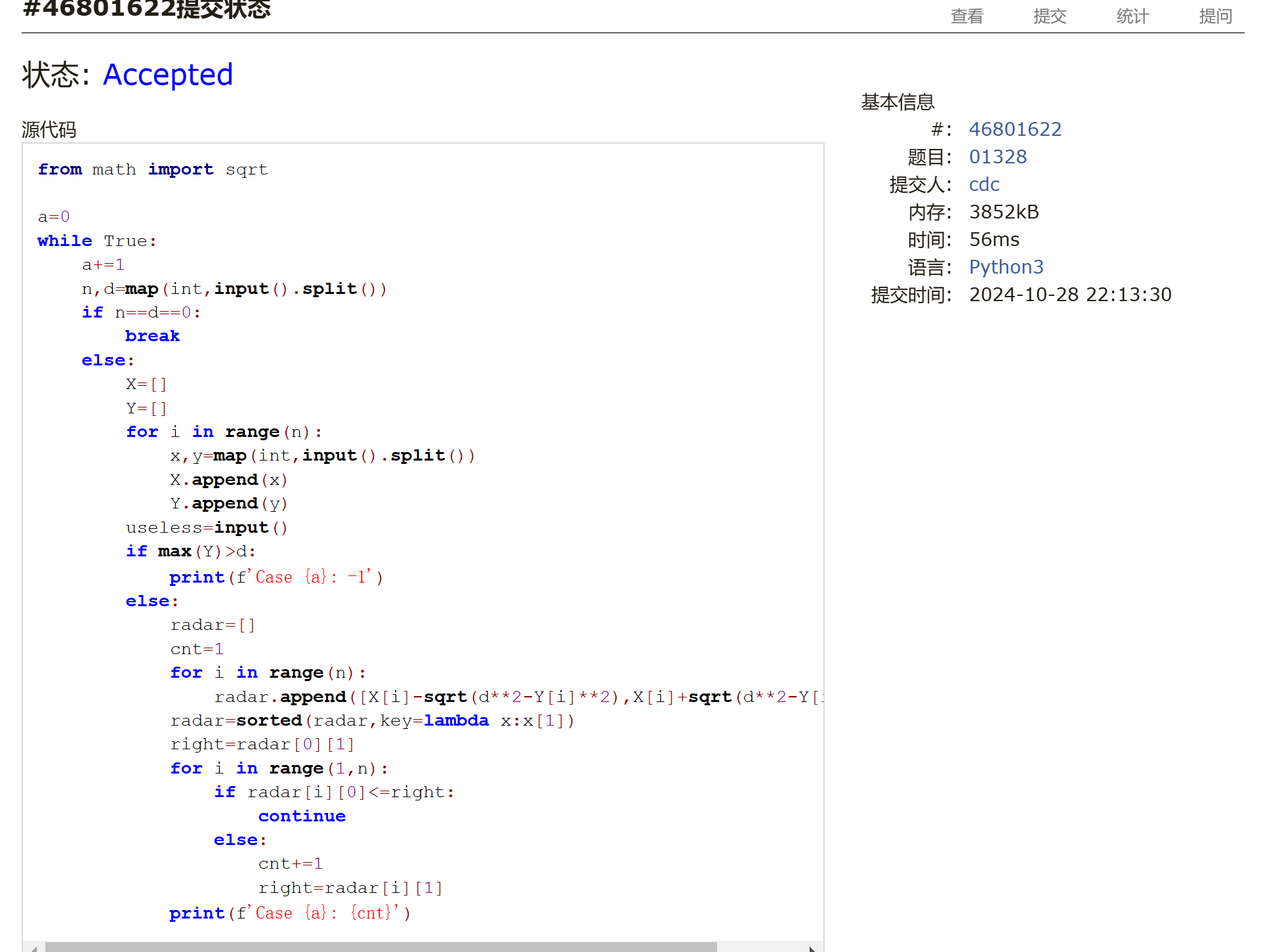
else:

cnt+=1

right=radar[i][1]

print(f'Case {a}: {cnt}')

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

## 2. 学习总结和收获

<mark>如果作业题目简单，有否额外练习题目，比如：OJ“计概 2024fall ”每日选做 、CF、LeetCode、洛谷

等网站题目。</mark>

感觉这次作业整体上没上次难，但是这次的坑好多啊，很不幸几乎全踩过:( 上周几乎没怎么做每日选做，这周争取多补点