

Assignment #5: Greedy穷举 Implementation

Updated 1939 GMT+8 Oct 21, 2024

2024 fall, Complied by 俞天麒 物理学院

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:

穷举，一直往d上加1，直到满足要求为止，对于每组数据时间复杂度也就O(n)，不会太大

代码:

```
k=1
while True:
    p,e,i,d=map(int,input().split())
    if p+e+i+d==4:
        break
    d0=d+1
    while True:
        if abs(d0-p)%23==0 and abs(d0-e)%28==0 and abs(d0-i)%33==0:
            print("Case "+str(k)+": the next triple peak occurs in "+str(d0-d)+" days.")
            break
        else:
            d0+=1
    k+=1
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
k=1
while True:
    p,e,i,d=map(int,input().split())
    if p+e+i+d==4:
        break
    d0=d+1
    while True:
        if abs(d0-p)%23==0 and abs(d0-e)%28==0 and abs(d0-i)%33==0:
            print("Case "+str(k)+": the next triple peak occurs in "+str(d0-d))
            break
        else:
            d0+=1
    k+=1
```

©2002-2022 POJ 京ICP备20010980号-1

基本信息

#: 46311705
题目: 04148
提交人: 24n2400011425
内存: 3584kB
时间: 40ms
语言: Python3
提交时间: 2024-10-05 00:17:53

English 帮助 关于

18211: 军备竞赛

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

思路:

要买肯定一直买最便宜的，然后买不起了再卖一个最贵的，然后再买。可以用双指针记录买到哪和卖到哪。但是注意while循环的返回条件，一个是都买完了，还有一个是最后一个买不起，但是也别卖了。

代码:

```
p=int(input())
a=sorted(list(map(int,input().split())))
i,j=0,len(a)-1
we,op=0,0
if p<a[0]:
    print(0)
    exit()
while i<=j:
    while p>=a[i]:
        p-=a[i]
        i+=1
        we+=1
    if i==len(a):
        break
    #print(i+1,j+1)
    if we>op and i<j:
        p+=a[j]
        j-=1
        op+=1
    if i==j and p<a[i]:
        break
print(we-op)
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
p=int(input())
a=sorted(list(map(int,input().split())))
i,j=0,len(a)-1
we,op=0,0
if p<a[0]:
    print(0)
    exit()
while i<=j:
    while p>=a[i]:
        p-=a[i]
        i+=1
        we+=1
    if i==len(a):
        break
    #print(i+1,j+1)
    if we>op and i<j:
        p+=a[j]
        j-=1
        op+=1
    if i==j and p<a[i]:
        break
print(we-op)
```

基本信息

#: 46648254
题目: 18211
提交人: 24n2400011425
内存: 3656kB
时间: 24ms
语言: Python3
提交时间: 2024-10-21 23:10:23

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

21554: 排队做实验

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

思路:

平均数等于总和除以人数，人数一定，总和要最小。排队越前面的人所花的时间要加越多次，所以应该让时间最少的人在前面。

代码:

```
n=int(input())
a=list(map(int,input().split()))
stu=[]
for i in range(n):
    stu.append([a[i],i+1])
stu.sort(key=lambda x:(x[0],x[1]))
for i in range(n):
    print(stu[i][1],end=" ")
print()
ans=0
for i in range(n):
    ans+=stu[i][0]**(n-i-1)
print("{:.2f}".format(ans/n))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n=int(input())
a=list(map(int,input().split()))
stu=[]
for i in range(n):
    stu.append([a[i],i+1])
stu.sort(key=lambda x:(x[0],x[1]))
for i in range(n):
    print(stu[i][1],end=" ")
print()
ans=0
for i in range(n):
    ans+=stu[i][0]*(n-i-1)
print("{:.2f}".format(ans/n))
```

基本信息

#: 46640510
题目: 21554
提交人: 24n2400011425
内存: 3648kB
时间: 23ms
语言: Python3
提交时间: 2024-10-21 17:57:55

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

01008: Maya Calendar

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

思路:

很恶心的枚举，要一个个检查自己打字有没有打错.....思路比较简单，总天数算出来，剩下的就是对应取两个余数就行了。注意13的倍数应该显示13，而不是除以13的余数0.还要注意总天数刚好整除260的情况

代码:

```
def judge(x):
    if x%13==0:
        return 13
    else:
        return x%13
haab=['pop', 'no', 'zip', 'zotz', 'tzec', 'xul', 'yoxkin', 'mol', 'chen',
'yax', 'zac', 'ceh', 'mac', 'kankin', 'muan', 'pax', 'koyab', 'cumhu', 'uayet']
tzolkin=['ahau','imix', 'ik', 'akbal', 'kan', 'chicchan', 'cimi', 'manik',
'lamat', 'muluk','ok', 'chuen', 'eb', 'ben', 'ix', 'mem', 'cib', 'caban',
'eznab', 'canac']
n=int(input())
print(n)
for _ in range(n):
    a,b,c=input().split()
    day=int(a[0:len(a)-1])
    year=int(c)
    month=haab.index(b)
    daytot=year*365+month*20+day+1
    #print(day,year,month,daytot)
    print(str(judge(daytot))+ " "+str(tzolkin[daytot%20])+ " "+str((daytot-
1)//260))
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
def judge(x):
    if x%13==0:
        return 13
    else:
        return x%13
haab=['pop', 'no', 'zip', 'zotz', 'tzec', 'xul', 'yoxkin', 'mol', 'chen', 'yax', 'tzolkin=['ahau','imix','ik','akbal','kan','chicchan','cimi','manik','lame
n=int(input())
print(n)
for _ in range(n):
    a,b,c=input().split()
    day=int(a[0:len(a)-1])
    year=int(c)
    month=haab.index(b)
    daytot=year*365+month*20+day+1
    #print(day,year,month,daytot)
    print(str(judge(daytot))+""+str(tzolkin[daytot%20])+""+str((daytot%20)%13))
```

©2002-2022 POJ 京ICP备20010980号-1

基本信息

#: 46573517
题目: 01008
提交人: 24n2400011425
内存: 3680kB
时间: 24ms
语言: Python3
提交时间: 2024-10-18 16:58:09

English 帮助 关于

545C. Woodcutters

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

思路:

我比较笨，只会暴力枚举.....

从左边往右边分析，首先最左边的可以往左边倒没话说。之后的树能往左边倒的只要往左边倒，对后续结果就不会产生影响，还能多一个倒下的树，所以就让他往左边倒。接下来考虑往右边的情况，如果下一个树很配合刚好能一起往中间倒不影响，那就直接分析在下一棵树，结果+2.如果都不能倒，那这棵树就别动了，直接到下一个。如果只能倒一个，那反正不浪费空间，就把能倒的倒了，如果是这棵树右倒，那分析下一个，如果是下一个左边倒，那再下一个。麻烦的是都可以的情况，反正这段路只能倒一个，这棵树是只能倒这里，下一棵说不定右边还有机会，那就把这段路留给这棵树。

代码:

```
a=[]
n=int(input())
for i in range(n):
    x,h=map(int,input().split())
    a.append([x,h])
ans=1
i=1
mark=0
while i<n-1:
    if a[i][0]-a[i-1][0]>a[i][1] and mark==0:
        ans+=1
        i+=1
    elif a[i+1][0]-a[i][0]>a[i][1]+a[i+1][1]:
        ans+=2
        i+=2
        mark=0
    elif a[i+1][0]-a[i][0]>a[i][1]:
        if a[i+1][0]-a[i][0]<=a[i+1][1]:
            ans+=1
            i+=1
            mark=0
```

```

else:
    ans+=1
    mark=1
    i+=1
else:
    i+=1
    mark=0
if i==n-1:
    print(ans+1)
else:
    print(ans)

```

代码运行截图 (至少包含有"Accepted")

ID	Editor	Problem	Lang	Verdict	Time	Memory	Date	Duration	Actions
286670355	Practice: Chriniko	545C - 14	Python 3	Accepted	390 ms	16628 KB	2024-10-19 11:44:10	2024-10-19 11:44:11	Compare

→ Source Copy

```

a=[]
n=int(input())
for i in range(n):
    x,h=map(int,input().split())
    a.append([x,h])
ans=1
i=1
mark=0
while i<n-1:
    if a[i][0]-a[i-1][0]>a[i][1] and mark==0:
        ans+=1
        i+=1
    elif a[i+1][0]-a[i][0]>a[i][1]+a[i+1][1]:
        ans+=2
        i+=2
        mark=0
    elif a[i+1][0]-a[i][0]>a[i][1]:
        if a[i+1][0]-a[i][0]<=a[i+1][1]:
            ans+=1
            i+=1
            mark=0
        else:
            ans+=1
            mark=1
            i+=1
    else:
        i+=1
        mark=0
if i==n-1:
    print(ans+1)
else:
    print(ans)

```

01328: Radar Installation

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

思路：

把点是否在以雷达为圆心的圆内转化为点是否在x轴 ($x-\sqrt{d^2-y^2}, x+\sqrt{d^2-y^2}$)上。之后这些点变成了一堆区间，要求找出最少的雷达使得每个区间至少有一个雷达。那就把这些区间按照结束位置排序。在前一个区间的结束位置放的雷达一定比下一个区间的结束位置在前面，所以只要下一个区间的开始位置小于雷达位置就行了。直到某一个区间的开始位置大于这个雷达，就意味着要多一个，放在这个区间的结束位置，这样直到把所有区间遍历完就好了。

代码：

```

k=1
while True:
    n,d=map(int,input().split())
    if n==d==0:
        break
    a=[]
    suc=False

```

```

for _ in range(n):
    x,y=map(int,input().split())
    a.append([x-(d**2-y**2)**0.5,x+(d**2-y**2)**0.5])
    if y>d:
        suc=True
blank=input()
if suc or d<0:
    print("Case "+str(k)+": -1")
    k+=1
    continue
a.sort(key=lambda x:x[1])
r=a[0][1]
ans=1
for i in range(1,n):
    if r<a[i][0]:
        r=a[i][1]
        ans+=1
print("Case "+str(k)+": "+str(ans))
k+=1

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

k=1
while True:
    n,d=map(int,input().split())
    if n==d==0:
        break
    a=[]
    suc=False
    for _ in range(n):
        x,y=map(int,input().split())
        a.append([x-(d**2-y**2)**0.5,x+(d**2-y**2)**0.5])
        if y>d:
            suc=True
    blank=input()
    if suc or d<0:
        print("Case "+str(k)+": -1")
        k+=1
        continue
    a.sort(key=lambda x:x[1])
    r=a[0][1]
    ans=1
    for i in range(1,n):
        if r<a[i][0]:
            r=a[i][1]
            ans+=1
    print("Case "+str(k)+": "+str(ans))
    k+=1

```

基本信息

#: 46588306
 题目: 01328
 提交人: 24n2400011425
 内存: 3744kB
 时间: 53ms
 语言: Python3
 提交时间: 2024-10-19 10:34:44

2. 学习总结和收获

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。

救救孩子吧.....使孩子的多少头发葬送，使孩子多少次情绪崩溃，就为了那一句ac，作为计概对我最美的告白.....

孩子为了ac一道题基本上都是啃1h以上.....特别是在现在基本上已经语法过关，但是模型没见多少的情况下。明白计概题目也要像物理竞赛一样刷题见模型，毕竟也有一个东西叫信息竞赛。完全陌生的模型需要从头开始看题解，需要我们去体会代码独特的思维方式，这一定得下大功夫的，真正的学习或许才刚刚开始。但是每次见过并熟悉一个新模型，下次再遇到的时候就能很熟练的运用了，说不定考试时候就遇到了呢 ((而且计算机的思维方式和物理图像的思维不是很一样，慢慢体会这之中的思维打开了脑洞，或许有一天能运用在物理之中吧，再不济就见见世面也是很有意思的。