

Assignment #B: 贪心、矩阵和动态规划

Updated 0118 GMT+8 Nov 21, 2023

2023 fall, Compiled by 同学的姓名、院系

说明:

本周作业留点难题，期中考试结束了，需要学习计算概论了。这次不分必做选做题目了，如果耗时太长，直接找答案看。两个题解，经常更新。所以最好从这个链接下载最新的，<https://github.com/GMyhf/2020fall-cs101>。

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, 学号），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。

2) 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、作业评论有md或者doc。

3) 如果不能在截止前提交作业，请写明原因。

编程环境

(请改为同学的操作系统、编程环境等)

操作系统: macOS Ventura 13.4.1 (c)

Python编程环境: Spyder IDE 5.2.2, PyCharm 2023.1.4 (Professional Edition)

C/C++编程环境: Mac terminal vi (version 9.0.1424), g++/gcc (Apple clang version 14.0.3, clang-1403.0.22.14.1)

1. 题目

如果耗时太长，直接看解题思路，或者源码

02786:Pell数列

<http://cs101.openjudge.cn/practice/02786/>

思路:

找到最大的序号，统一生成数列，减少计算次数

代码

```
1  '''
2  刘思瑞 2100017810
3  '''
4  def find(n):
5      m = [1,2]
6      for i in range(2,n):
7          m.append((2*m[i-1]+m[i-2])%32767)
8      return m
9
10 li = []
11 num = int(input())
12 for i in range(num):
13     li.append(int(input()))
14 n = max(li)
15 m = find(n)
16 for i in li:
17     print(m[i-1])
```

代码运行截图

状态: Accepted

源代码

```
'''
刘思瑞 2100017810
'''
def find(n):
    m = [1,2]
    for i in range(2,n):
        m.append((2*m[i-1]+m[i-2])%32767)
    return m

li = []
num = int(input())
for i in range(num):
    li.append(int(input()))
n = max(li)
m = find(n)
for i in li:
    print(m[i-1])
```

04133:垃圾炸弹

matrices, <http://cs101.openjudge.cn/practice/04133/>

思路:

遍历

代码

```
1  '''
2  刘思瑞 2100017810
3  '''
4
5  li , xli ,yli = [],[],[]
6  maxx = 0
7  times = 0
8  for i in range(1025):
9      li.append([0]*1025)
10 d = int(input())
11 num = int(input())
12 for g in range(num):
13     x,y,i = map(int,input().split())
14     for j in range(max(0,y-d),min(1025,y+d+1)):
15         for k in range(max(0,x-d),min(1025,x+d+1)):
16             li[j][k] += i
17 for i in range(1025):
18     for j in range(1025):
19         if li[i][j] >maxx:
20             maxx = li[i][j]
21             times = 1
22         elif li[i][j] == maxx:
23             times+=1
24 print(times,maxx)
```

代码运行截图

状态: Accepted

源代码

```
'''
刘思瑞 2100017810
'''

li , xli ,yli = [],[],[]
maxx = 0
times = 0
for i in range(1025):
    li.append([0]*1025)
d = int(input())
num = int(input())
for g in range(num):
    x,y,i = map(int,input().split())
    for j in range(max(0,y-d),min(1025,y+d+1)):
        for k in range(max(0,x-d),min(1025,x+d+1)):
            li[j][k] += i
for i in range(1025):
    for j in range(1025):
        if li[i][j] >maxx:
            maxx = li[i][j]
            times = 1
        elif li[i][j] == maxx:
            times+=1
print(times,maxx)
```

26971:分发糖果

greedy, <http://cs101.openjudge.cn/routine/26971/>

思路:

按一个上升下降序列为周期计算

代码

```
1 '''
2 刘思瑞 2100017810
3 '''
4 def count(i,j,k):
5     maxx = max(i,j)
6     minn = min(i,j)
7     if k:
8         if maxx > minn:
9             maxx-=1
10            minn+=1
11            return minn*(minn+1)//2+(maxx+1)*(maxx+2)//2 -1
12
13 num = int(input())
```

```

14 li = []
15 li += list(map(int,input().split()))
16 li.append(-1)
17 flag = True
18 upnum = -1
19 downnum = 1
20 equal = False
21 sum = 0
22 for i in range(num):
23     if flag:
24         if li[i] < li[i-1]:
25             flag = not flag
26         elif li[i] == li[i-1]:
27             if li[i] == li[i+1]:
28                 sum += 1
29                 continue
30             flag = not flag
31             equal = True
32     else:
33         upnum += 1
34     else:
35         if li[i] > li[i-1]:
36             flag = not flag
37             sum += count(upnum, downnum, equal)
38             equal = False
39             upnum = 1
40             downnum = 1
41         elif li[i] == li[i-1]:
42             if li[i] == li[i+1]:
43                 sum += 1
44                 continue
45             flag = not flag
46             sum += count(upnum, downnum, equal)
47             equal = False
48             upnum = 0
49             sum += 1
50             downnum = 1
51         else:
52             downnum += 1
53     if flag:
54         sum += count(upnum, 0, equal)
55     else:
56         sum += count(upnum, downnum, equal)
57 print(sum+1)

```

代码运行截图

状态: Accepted

源代码

```
'''
刘思瑞 2100017810
'''
def count(i,j,k):
    maxx = max(i,j)
    minn = min(i,j)
    if k:
        if maxx > minn:
            maxx-=1
            minn+=1
    return minn*(minn+1)//2+(maxx+1)*(maxx+2)//2 -1

num = int(input())
li = []
li += list(map(int,input().split()))
li.append(-1)
flag = True
upnum = -1
downnum = 1
equal = False
sum = 0
for i in range(num):
    if flag:
        if li[i]<li[i-1]:
            flag = not flag
        elif li[i] == li[i-1]:
            if li[i] == li[i+1]:
                sum+=1
                continue
```

26976:摆动序列

greedy, <http://cs101.openjudge.cn/routine/26976/>

思路:

找单调序列

代码

```
1  '''
2  刘思瑞 2100017810
3  '''
4  n = int(input())
5  li = list(map(int,input().split()))
6  num = 1
7  flag = True
8  for i in range(n-1):
9      if li[i] !=li[i+1]:
```

```

10         flag = li[i+1]>li[i]
11         num=2
12         break
13
14     for i in range(n-1):
15         if li[i+1] == li[i]:
16             continue
17         if flag:
18             if li[i+1] < li[i]:
19                 num+=1
20                 flag = not flag
21         else:
22             if li[i+1] > li[i]:
23                 num+=1
24                 flag = not flag
25
26     print(num)

```

代码运行截图

状态: Accepted

源代码

```

'''
刘思瑞 2100017810
'''
n = int(input())
li = list(map(int,input().split()))
num = 1
flag = True
for i in range(n-1):
    if li[i] !=li[i+1]:
        flag = li[i+1]>li[i]
        num=2
        break

for i in range(n-1):
    if li[i+1] == li[i]:
        continue
    if flag:
        if li[i+1] < li[i]:
            num+=1
            flag = not flag
    else:
        if li[i+1] > li[i]:
            num+=1
            flag = not flag

print(num)

```

27104:世界杯只因

<http://cs101.openjudge.cn/practice/27104/>

思路:

贪心

代码

```
1  '''
2  2100017810 刘思瑞
3  '''
4  def find(n, li, long, number):
5      for i in range(min(0, long//2-1), num):
6          if i-li[i] <= n and i+li[i] >= n:
7              if i+li[i] > long:
8                  long = i + li[i]
9              number+=1
10         return long, li, long, number
11 num = int(input())
12 number = 0
13 li = list(map(int, input().split()))
14 long = 0
15 long, li, long, number = find(0, li, long, number)
16 while True:
17     if long>=num-1:
18         break
19     long, li, long, number = find(long+1, li, long, number)
20 print(number)
```

代码运行截图

状态: Accepted

基

源代码

```
'''
2100017810 刘思瑞
'''
def find(n, li, long, number):
    for i in range(min(0, long//2-1), num):
        if i-li[i] <= n and i+li[i] >= n:
            if i+li[i] > long:
                long = i + li[i]
            number+=1
    return long, li, long, number
num = int(input())
number = 0
li = list(map(int, input().split()))
long = 0
long, li, long, number = find(0, li, long, number)
while True:
    if long>=num-1:
        break
    long, li, long, number = find(long+1, li, long, number)
print(number)
```

主

CF1000B: Light It Up

greedy, 1500, <https://codeforces.com/problemset/problem/1000/B>

思路:

贪心, 只在最接近的地方插入

代码

```
1 '''
2 刘思瑞 2100017810
3 '''
4 n,m = map(int,input().split())
5 li = [0]
6 li += list(map(int,input().split())) + [m]
7 atime,parttime,maxtime=0,0,0
8 for i in range(n//2+1):
9     atime+=li[2*i+1] - li[2*i]
10    maxtime = atime
11    for i in range(n//2+1):
12        parttime += li[2*i+1] - li[2*i]
13        maxtime = max(max(m-li[2*i+1]-1-atime+parttime,atime-
14        parttime)+parttime,maxtime)
15    print(maxtime)
```

代码运行截图

By meinvader, contest: Educational Codeforces Round 46 (Rated for Div. 2), problem: (B) Light It Up, **Accepted**, 2020-08-17 14:00:00

```
'''
刘思瑞 2100017810
'''
n,m = map(int,input().split())
li = [0]
li += list(map(int,input().split())) + [m]
atime,parttime,maxtime=0,0,0
for i in range(n//2+1):
    atime+=li[2*i+1] - li[2*i]
maxtime = atime
for i in range(n//2+1):
    parttime += li[2*i+1] - li[2*i]
    maxtime = max(max(m-li[2*i+1]-1-atime+parttime, atime-parttime)+parttime, maxtime)
print(maxtime)
```

1. Judgement Protocol

2. 学习总结和收获

这次的作业都有思路，但是感觉由于思路很复杂所以程序出了很多bug，本来能按时完成的，但是世界杯的题目一直找不到错误，后来发现是因为跳出循环的条件错了（（还有糖果的题目，在题解上看到了很聪明的方法，按照我的方法会多出很多关于相等元素的探讨，也是在在这个地方出了好多bug。。