Assignment #B: 贪心、矩阵和动态规划

Updated 0118 GMT+8 Nov 21, 2023

2023 fall, Complied 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

```
'''
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])
```

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

```
'''
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)
```

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

```
'''
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
            flag = not flag
            equal = True
        else:
            upnum += 1
    else:
        if li[i]>li[i-1]:
            flag = not flag
            sum += count(upnum,downnum,equal)
            equal = False
            upnum = 1
            downnum =1
        elif li[i] == li[i-1]:
            if li[i] == li[i+1]:
                sum+=1
                continue
            flag = not flag
            sum += count(upnum,downnum,equal)
            equal = False
            upnum = 0
            sum+= 1
            downnum =1
        else:
            downnum += 1
if flag:
    sum += count(upnum,0,equal)
else:
    sum += count(upnum,downnum,equal)
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

```
'''
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)
```

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

```
'''
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)
```

代码运行截图

状态：Accepted

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)
```

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

```
'''
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)
```

代码运行截图

By meinvader, contest: Educational Codeforces Round 46 (Rated for Div. 2), problem: (B) Light It Up, **Accepted**,

```
'''
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)
```

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。。