Assignment #8: Nov 月考

Updated 1355 GMT+8 Nov 2, 2023

2023 fall, Complied by 同学的姓名、院系

说明：

1）1）Nov月考： AC6（请改为同学的通过数） 。题目都在"练习"里面，按照数字题号能找到，可以重新提交。作业中提交自己最满意版本的代码和截图。

2）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。

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、作业评论有md或者doc。

4）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. 必做题目

23563: 多项式时间复杂度

string/implementation/math, http://cs101.openjudge.cn/practice/23563

思路：

善用split函数

代码

```
'''
刘思瑞 2100017810
'''
max = 0
s = list(input().split('+'))
for i in s:
    j = list(i.split('n^'))
    if j[0] != '0':
        if int(j[1]) > max:
            max = int(j[1])
print('n^'+str(max))
```

代码运行截图

03143: 验证“歌德巴赫猜想”

math, http://cs101.openjudge.cn/practice/03143

思路:

遍历

代码

```
'''
刘思瑞 2100017810
'''
def su(i):
    for j in range(2,int(i**0.5)+2):
        if i%j == 0:
            return False
    return True
def find(n):
    if n < 6 or n % 2 != 0 :
        print('Error!')
        return
    for i in range(3,n//2 +1 ,2):
        if su(i):
            if su(n-i):
                print(str(n)+ '=' + str(i) + '+' + str(n-i))
    return
n = int(input())
find(n)
```

代码运行截图

状态: Accepted

源代码

```
'''
刘思瑞 2100017810
'''
def su(i):
    for j in range(2,int(i**0.5)+2):
        if i%j == 0:
            return False
    return True
def find(n):
    if n < 6 or n % 2 != 0 :
        print('Error!')
        return
    for i in range(3,n//2 +1 ,2):
        if su(i):
            if su(n-i):
                print(str(n)+ '=' + str(i) + '+' + str(n-i))
    return
n = int(input())
find(n)
```

23566: 决战双十一

implementation, http://cs101.openjudge.cn/practice/23566

思路：

用数组储存

代码

```
'''
刘思瑞 2100017810
'''

n , m = map(int,input().split())
store = [0]*m
totalyouhui = 0
for i in range(n):
    inde , price = map(int,input().split())
    store[inde - 1] += price
for i in range(m):
    manjian , youhui = map(int,input().split('-'))
    if store[i] >= manjian:
        totalyouhui += youhui
totalyouhui += ((sum(store))//200)*30
print(sum(store)-totalyouhui)
```

代码运行截图

状态：Accepted

源代码

```
'''
刘思瑞 2100017810
'''

n , m = map(int,input().split())
store = [0]*m
totalyouhui = 0
for i in range(n):
    inde , price = map(int,input().split())
    store[inde - 1] += price
for i in range(m):
    manjian , youhui = map(int,input().split('-'))
    if store[i] >= manjian:
        totalyouhui += youhui
totalyouhui += ((sum(store))//200)*30
print(sum(store)-totalyouhui)
```

03670: 计算鞍点

matrice, http://cs101.openjudge.cn/practice/03670

思路:

直接遍历即可

代码

```
'''
刘思瑞 2100017810
'''
matrix = []
m = 0
for i in range(5):
    matrix.append(list(map(int,input().split())))
for i in range(5):
    inde = matrix[i].index(max(matrix[i]))
    flag = 1
    for j in matrix:
        if j[inde] < max(matrix[i]):
            flag = 0
            break
    if flag == 1:
        print(i+1,matrix[i].index(max(matrix[i]))+1,max(matrix[i]))
        m = 1
if m == 0:
    print('not found')
```

代码运行截图

状态: Accepted

源代码

```
'''
刘思瑞 2100017810
'''
matrix = []
m = 0
for i in range(5):
    matrix.append(list(map(int,input().split())))
for i in range(5):
    inde = matrix[i].index(max(matrix[i]))
    flag = 1
    for j in matrix:
        if j[inde] < max(matrix[i]):
            flag = 0
            break
    if flag == 1:
        print(i+1,matrix[i].index(max(matrix[i]))+1,max(matrix[i]))
        m = 1
if m == 0:
    print('not found')
```

19948: 因材施教

greedy, http://cs101.openjudge.cn/practice/19948

思路:

分组只关注最值，可以直接从分组的位置开始找

代码

```
'''
刘思瑞 2100017810
'''
n , m = map(int,input().split())
grade = list(map(int,input().split()))
grade.sort()
minusgrade = []
separa = [0]
sum = grade[-1] - grade[0]
for i in range(n-1):
    minusgrade.append(grade[i+1]-grade[i])
minusgrade.sort(reverse=True)
for i in range(m-1):
    sum -= minusgrade[i]
print(sum)
```

代码运行截图

状态：Accepted

源代码

```
'''
刘思瑞 2100017810
'''
n , m = map(int,input().split())
grade = list(map(int,input().split()))
grade.sort()
minusgrade = []
separa = [0]
sum = grade[-1] - grade[0]
for i in range(n-1):
    minusgrade.append(grade[i+1]-grade[i])
minusgrade.sort(reverse=True)
for i in range(m-1):
    sum -= minusgrade[i]
print(sum)
```

18182: 打怪兽

implementation/sortings/data structures, http://cs101.openjudge.cn/practice/18182/

思路：

用两个数组分别存储时间和伤害，冒泡排序再对伤害排序即可

代码

```
'''
刘思瑞 2100017810
'''
testnum = int(input())
for i in range(testnum):
    n ,m ,b = map(int,input().split())
    release = []
    time = []
    flag = 1
    for j in range(n):
        t , hurt = map(int,input().split())
        if t in time:
            release[time.index(t)].append(hurt)
        else:
            time.append(t)
            release.append([hurt])
    l = len(time)
    for j in range(l-1):
        for k in range(l-1-j):
```

```
            if time[k] > time[k+1]:
                time[k] , time[k+1] , release[k] , release[k+1] = time[k+1]
, time[k] , release[k+1] , release[k]
    for j in range(l):
        release[j].sort(reverse = True)
        hurting = sum(release[j][:m])
        if hurting >= b:
            print(time[j])
            flag = 0
            break
        else:
            b -= hurting
    if flag:
        print('alive')
```

代码运行截图

状态：Accepted

源代码

```
'''
刘思瑞 2100017810
'''
testnum = int(input())
for i in range(testnum):
    n ,m ,b = map(int,input().split())
    release = []
    time = []
    flag = 1
    for j in range(n):
        t , hurt = map(int,input().split())
        if t in time:
            release[time.index(t)].append(hurt)
        else:
            time.append(t)
            release.append([hurt])
    l = len(time)
    for j in range(l-1):
        for k in range(l-1-j):
            if time[k] > time[k+1]:
                time[k] , time[k+1] , release[k] , release[k+1] = time[
    for j in range(l):
        release[j].sort(reverse = True)
        hurting = sum(release[j][:m])
        if hurting >= b:
            print(time[j])
            flag = 0
            break
        else:
            b -= hurting
    if flag:
        print('alive')
```

2. 学习总结和收获

依然期中周，下次补上。。