

Assignment #8: Nov 月考

Updated 1355 GMT+8 Nov 2, 2023

2023 fall, Compiled 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

```
1 '''
2 刘思瑞 2100017810
3 '''
4 max = 0
5 s = list(input().split('+'))
6 for i in s:
7     j = list(i.split('n^'))
8     if j[0] != '0':
9         if int(j[1]) > max:
10             max = int(j[1])
11 print('n^'+str(max))
```

代码运行截图

状态: Accepted

源代码

```
'''
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))
```

03143: 验证“歌德巴赫猜想”

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

思路:

遍历

代码

```
1  '''
2  刘思瑞 2100017810
3  '''
4  def su(i):
5      for j in range(2,int(i**0.5)+2):
6          if i%j == 0:
7              return False
8      return True
9  def find(n):
10     if n < 6 or n % 2 != 0 :
11         print('Error!')
12         return
13     for i in range(3,n//2 +1 ,2):
14         if su(i):
15             if su(n-i):
16                 print(str(n)+ '=' + str(i) + '+' + str(n-i))
17     return
18 n = int(input())
19 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

```
1  '''
2  刘思瑞 2100017810
3  '''
4
5  n , m = map(int,input().split())
6  store = [0]*m
7  totalyouhui = 0
8  for i in range(n):
9      inde , price = map(int,input().split())
10     store[inde - 1] += price
11  for i in range(m):
12     manjian , youhui = map(int,input().split('-'))
13     if store[i] >= manjian:
14         totalyouhui += youhui
15  totalyouhui += ((sum(store))/200)*30
16  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

```
1  '''
2  刘思瑞 2100017810
3  '''
4  matrix = []
5  m = 0
6  for i in range(5):
7      matrix.append(list(map(int, input().split())))
8  for i in range(5):
9      inde = matrix[i].index(max(matrix[i]))
10     flag = 1
11     for j in matrix:
12         if j[inde] < max(matrix[i]):
13             flag = 0
14             break
15     if flag == 1:
16         print(i+1, matrix[i].index(max(matrix[i]))+1, max(matrix[i]))
17         m = 1
18 if m == 0:
19     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

```
1  '''
2  刘思瑞 2100017810
3  '''
4  n, m = map(int, input().split())
5  grade = list(map(int, input().split()))
6  grade.sort()
7  minusgrade = []
8  separa = [0]
9  sum = grade[-1] - grade[0]
10 for i in range(n-1):
11     minusgrade.append(grade[i+1]-grade[i])
12 minusgrade.sort(reverse=True)
13 for i in range(m-1):
14     sum -= minusgrade[i]
15 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

```
1  '''
2  刘思瑞 2100017810
3  '''
4  testnum = int(input())
5  for i in range(testnum):
6      n ,m ,b = map(int,input().split())
7      release = []
8      time = []
9      flag = 1
10     for j in range(n):
11         t , hurt = map(int,input().split())
12         if t in time:
13             release[time.index(t)].append(hurt)
14         else:
15             time.append(t)
16             release.append([hurt])
17     l = len(time)
18     for j in range(l-1):
19         for k in range(l-1-j):
```

```

20         if time[k] > time[k+1]:
21             time[k] , time[k+1] , release[k] , release[k+1] = time[k+1]
, time[k] , release[k+1] , release[k]
22     for j in range(1):
23         release[j].sort(reverse = True)
24         hurting = sum(release[j][:m])
25         if hurting >= b:
26             print(time[j])
27             flag = 0
28             break
29         else:
30             b -= hurting
31     if flag:
32         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[k+1]
, time[k] , release[k+1] , release[k]
    for j in range(1):
        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。。