

Assignment #4: T-primes + 贪心

Updated 0337 GMT+8 Oct 15, 2024

2024 fall, Compiled by 陈张涵、工学院

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) 课程网站是Canvas平台, <https://pku.instructure.com>, 学校通知9月19日导入选课名单后启用。作业写好后, 保留在自己手中, 待9月20日提交。

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

34B. Sale

greedy, sorting, 900, <https://codeforces.com/problemset/problem/34/B>

思路: 只买价格为负数的电视机

代码

```
n,m=map(int,input().split())
list1=[]
a=0
money = 0
list1 = list(map(int,input().split()))
list1.sort()
k = 0
while m > 0 and list1[k] < 0 and k <= len(list1)-1:
    money -= list1[k]
    m -= 1
    k += 1
print(money)
```

代码运行截图 (至少包含有"Accepted")

General									
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged	
280887886	Practice: chaain	34B - 4	Python 3	Accepted	154 ms	12 KB	2024-09-12 19:29:02	2024-09-12 19:29:02	<button>Compare</button>

→ Source
Copy

```

n,m=map(int,input().split())
list1=[]
a=0
money = 0
list1 = list(map(int,input().split()))
list1.sort()
k = 0
while n > 0 and list1[k] < 0 and k<len(list1)-1:
    money -= list1[k]
    n -= 1
    k += 1
print(money)

```

160A. Twins

greedy, sortings, 900, <https://codeforces.com/problemset/problem/160/A>

思路：尽量拿面值最大的硬币

代码

```

n = int(input())
list1 =list(map(int,input().split()))
list1.sort()
sum = 0
for i in list1:
    sum += i
my = 0
piece = 0
for i in range(len(list1)):
    if my <=sum/2:
        my += list1[-1-i]
        piece += 1
print(piece)

```

代码运行截图 == (至少包含有"Accepted") ==

General									
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged	
280961853	Practice: chaain	160A - 20	Python 3	Accepted	124 ms	16 KB	2024-09-13 12:26:22	2024-09-13 12:26:22	<button>Compare</button>

→ Source
Copy

```

n = int(input())
list1 =list(map(int,input().split()))
list1.sort()
sum = 0
for i in list1:
    sum += i
my = 0
piece = 0
for i in range(len(list1)):
    if my <=sum/2:
        my += list1[-1-i]
        piece += 1
print(piece)

```

[Click](#) to see test details

1879B. Chips on the Board

constructive algorithms, greedy, 900, <https://codeforces.com/problemset/problem/1879/B>

思路：首先很容易证明一定满足所有行都放一个或所有列都放一个两个条件中的一个，接下来分别记算放满数值最小的行和数值最小的列所对应的值，取更小的那个

代码

```
cons1 = 0
cons2 = 0
ans = []
def mincost(m, list1, list2):
    sum1 = 0
    sum2 = 0
    list1.sort()
    list2.sort()
    for i in list1:
        sum1 += i
    for i in list2:
        sum2 += i
    cons1 = sum1 + m*list2[0]
    cons2 = sum2 + m*list1[0]
    return min(cons1, cons2)
n = int(input())
for i in range(n):
    k = int (input())
    a=list(map(int, input().split()))
    b=list(map(int, input().split()))
    ans.append(mincost(k, a, b))
for i in ans:
    print(i)
```

代码运行截图 (至少包含有"Accepted")

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
281126139	Practice: chaain	1879B - 10	Python 3	Accepted	452 ms	49684 KB	2024-09-14 16:13:05	2024-09-14 16:13:05	☆	<button>Compare</button>

Source

```
cons1 = 0
cons2 = 0
ans = []
def mincost(m, list1, list2):
    sum1 = 0
    sum2 = 0
    list1.sort()
    list2.sort()
    for i in list1:
        sum1 += i
    for i in list2:
        sum2 += i
    cons1 = sum1 + m*list2[0]
    cons2 = sum2 + m*list1[0]
    return min(cons1, cons2)
n = int(input())
for i in range(n):
    k = int (input())
    a=list(map(int, input().split()))
    b=list(map(int, input().split()))
    ans.append(mincost(k, a, b))
for i in ans:
    print(i)
```

[Click to see test details](#)

158B. Taxi

*special problem, greedy, implementation, 1100, <https://codeforces.com/problemset/problem/158/B>

思路：和堆箱子差不多

代码

```
import math
n = int(input())
groups=list(map(int,input().split()))
taxis=0
a1=groups.count(1)
a2=groups.count(2)
a3=groups.count(3)
a4=groups.count(4)
taxis+=(a4+a3)
rest=[0,2]
taxis+=math.ceil(a2/2)
buffer=rest[a2%2]+a3
if buffer<a1:
    taxis+=math.ceil((a1-buffer)/4)
print(taxis)
```

代码运行截图 (至少包含有"Accepted")

General										
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
286677935	Practice: chaain	158B - 10	PyPy 3-64	Accepted	248 ms	12588 KB	2024-10-19 12:33:13	2024-10-19 12:33:13	★	Compare

[→ Source](#) [Copy](#)

```
import math
n = int(input())
groups=list(map(int,input().split()))
taxis=0
a1=groups.count(1)
a2=groups.count(2)
a3=groups.count(3)
a4=groups.count(4)
taxis+=(a4+a3)
rest=[0,2]
taxis+=math.ceil(a2/2)
buffer=rest[a2%2]+a3
if buffer<a1:
    taxis+=math.ceil((a1-buffer)/4)
print(taxis)
```

[Click](#) to see test details

*230B. T-primes (选做)

binary search, implementation, math, number theory, 1300, <http://codeforces.com/problemset/problem/230/B>

思路：其实判断出是素数的三次方并不难，只要用到一点数论知识即可，

关键是如何高效判断素数

这里自学了欧拉筛

代码

```
from math import sqrt
def euler_sieve(n):
    is_prime = [True] * (n + 1)
    is_prime[0] = False
    is_prime[1] = False
    primes = []
    for i in range(2, n + 1):
        if is_prime[i]:
            primes.append(i)
            for j in range(i * i, n + 1, i):
                is_prime[j] = False
    return is_prime

list1 = euler_sieve(1000000)
n = int(input())
list2= map(int, input().split())
lsit3=[]
for i in list2:
    if int(sqrt(i))*2 == i:
        if list1[int(sqrt(i))]==True:
            lsit3.append('YES')
        else:
            lsit3.append('NO')
    else:
        lsit3.append('NO')
for i in lsit3:
    print(i)
```

代码运行截图 (至少包含有"Accepted")

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
284412144	Practice: chaaln	230B - 28	Python 3	Accepted	1124 ms	18336 KB	2024-10-05 10:24:40	2024-10-05 10:24:40		<button>Compare</button>

Source

```
from math import sqrt
def euler_sieve(n):
    is_prime = [True] * (n + 1)
    is_prime[0] = False
    is_prime[1] = False
    primes = []
    for i in range(2, n + 1):
        if is_prime[i]:
            primes.append(i)
            for j in range(i * i, n + 1, i):
                is_prime[j] = False
    return is_prime

list1 = euler_sieve(1000000)
n = int(input())
list2= map(int, input().split())
lsit3=[]
for i in list2:
    if int(sqrt(i))*2 == i:
        if list1[int(sqrt(i))]==True:
            lsit3.append('YES')
        else:
            lsit3.append('NO')
    else:
        lsit3.append('NO')
for i in lsit3:
    print(i)
```

[Click to see test details](#)

*12559: 最大最小整数 (选做)

greedy, strings, sortings, <http://cs101.openjudge.cn/practice/12559>

思路：用冒泡是肯定可以做的，只要比较字典序的大小就行了

但是想着能不能提一下速度，比方先用.sort按照字典序大概排一下序，但并没有什么有效思路

代码

```
n= int(input())
list1 = list(map(str, input().split()))
list2=list1.copy()
list3=list1.copy()
list2.sort(reverse=False)
list3.sort(reverse=True)
minv=''
maxv=''
for i in range(1, len(list2)):
    for j in range(0, len(list2)-i):
        if list2[j]+list2[j+1] > list2[j+1]+list2[j]:
            list2[j], list2[j + 1] = list2[j + 1], list2[j]
for i in range(1, len(list3)):
    for j in range(0, len(list3)-i):
        if list3[j]+list3[j+1] < list3[j+1]+list3[j]:
            list3[j], list3[j + 1] = list3[j + 1], list3[j]
for i in list2:
    minv+=i
for i in list3:
    maxv+=i

print(maxv,minv,sep=' ')
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n= int(input())
list1 = list(map(str, input().split()))
list2=list1.copy()
list3=list1.copy()
list2.sort(reverse=False)
list3.sort(reverse=True)
minv=''
maxv=''
for i in range(1, len(list2)):
    for j in range(0, len(list2)-i):
        if list2[j]+list2[j+1] > list2[j+1]+list2[j]:
            list2[j], list2[j + 1] = list2[j + 1], list2[j]
for i in range(1, len(list3)):
    for j in range(0, len(list3)-i):
        if list3[j]+list3[j+1] < list3[j+1]+list3[j]:
            list3[j], list3[j + 1] = list3[j + 1], list3[j]
for i in list2:
    minv+=i
for i in list3:
    maxv+=i

print(maxv,minv,sep=' ')
```

基本信息

#: 46600307
题目: 12559
提交人: 24n2400010996
内存: 3668kB
时间: 287ms
语言: Python3
提交时间: 2024-10-19 18:41:20

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。

本周作业有不少贪心算法的题型

t-primes是之前做的题目，主要是要自学判断素数的一些方法，题目本身其实难度不大

为了做最后一题去自学了一下冒泡排序

本周主要是在做oj上的贪心题，感觉正在慢慢体悟贪心题的一些思想方法，但要想真正进步还需要一定的时间和刷题量。目前正在积累方法和做题经验中

感觉有些题目不会做，稍微看看题解的提示也可能尝试写出代码，所以要尽量多从题解里学东西吧。