Assignment #4: T-primes + 贪心

Updated 0337 GMT+8 Oct 15, 2024

2024 fall, Complied by 徐贤天，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：

代码

```
#
n, m = map(int, input().split())
prices = list(map(int,input().split()))
prices.sort()
minus_price_num = 0
earning = 0
for i in range(n):
    if prices[i] < 0:
        minus_price_num += 1
    else:
        break
if minus_price_num <= m:
    for i in range(minus_price_num):
        earning += prices[i]
else:
    for i in range(m):
        earning += prices[i]
print(earning * (-1))
```

代码运行截图 （至少包含有"Accepted"）

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
281638032	Practice: Aunixt	34B - 4	Python 3	Accepted	154 ms	20 KB	2024-09-17 19:17:50	2024-09-17 19:17:51		Compare

→ Source Copy

```
n, m = map(int, input().split())
prices = list(map(int,input().split()))
prices.sort()
minus_price_num = 0
earning = 0
for i in range(n):
    if prices[i] < 0:
        minus_price_num += 1
    else:
        break
if minus_price_num <= m:
    for i in range(minus_price_num):
        earning += prices[i]
else:
    for i in range(m):
        earning += prices[i]
print(earning * (-1))
```

160A. Twins

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

思路：

代码

```
n = int(input())
coins = list(map(int, input().split()))
total = sum(coins)
my_money = 0
coins.sort(reverse=True)
for i in range(n):
    my_money += coins[i]
    if my_money > total - my_money:
        break
print(i + 1)
```

代码运行截图 ==（至少包含有"Accepted"）==

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS ROOM STANDINGS CUSTOM INVOCATION

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
281685225	Practice: Aunixt	160A - 20	Python 3	Accepted	154 ms	24 KB	2024-09-18 06:42:20	2024-09-18 06:42:20		Compare

→ Source Copy

```
n = int(input())
coins = list(map(int, input().split()))
total = sum(coins)
my_money = 0
coins.sort(reverse=True)
for i in range(n):
    my_money += coins[i]
    if my_money > total - my_money:
        break
print(i + 1)
```

1879B. Chips on the Board

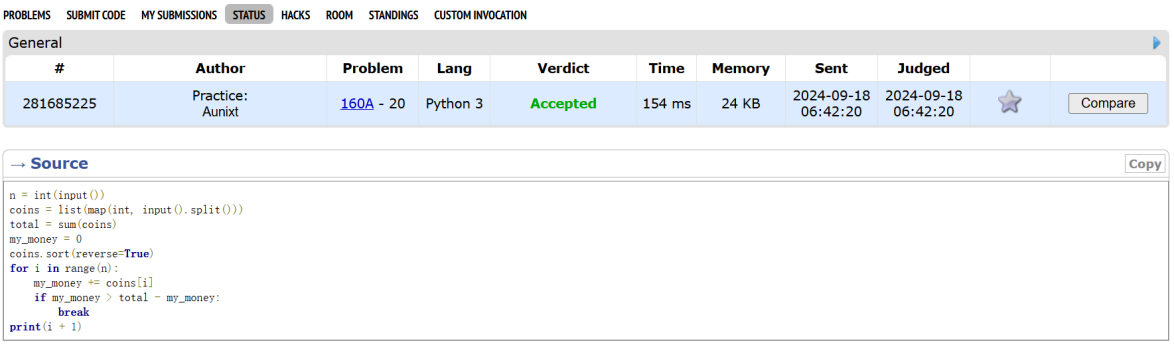
constructive algorithms, greedy, 900, https://codeforces.com/problemset/problem/1879/B

思路：

代码

```
n = int(input())
coins = list(map(int, input().split()))
total = sum(coins)
my_money = 0
coins.sort(reverse=True)
for i in range(n):
    my_money += coins[i]
    if my_money > total - my_money:
        break
print(i + 1)
```

代码运行截图 （至少包含有"Accepted"）


PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS ROOM STANDINGS CUSTOM INVOCATION

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
281685225	Practice: Aunixt	160A - 20	Python 3	Accepted	154 ms	24 KB	2024-09-18 06:42:20	2024-09-18 06:42:20		Compare

→ Source Copy

```
n = int(input())
coins = list(map(int, input().split()))
total = sum(coins)
my_money = 0
coins.sort(reverse=True)
for i in range(n):
    my_money += coins[i]
    if my_money > total - my_money:
        break
print(i + 1)
```

158B. Taxi

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

思路：

代码

```
import math
n = int(input())
ls = list(map(int, input().split()))
a, b, c, d = 0, 0, 0, 0
car_num = 0
for i in ls:
    if i == 1:
        a += 1
    elif i == 2:
```

```
        b += 1
    elif i == 3:
        c += 1
    elif i == 4:
        d += 1

if c >= a:
    car_num = d + math.ceil(b/2) + c
else:
    if b % 2 == 0:
        car_num = d + c + b//2 +math.ceil((a-c)/4)
    else:
        car_num = d + c + math.ceil(b/2) + math.ceil((a-c-2)/4)
print(car_num)
```

代码运行截图（至少包含有"Accepted"）

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS STANDINGS CUSTOM INVOCATION

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
286337425	Practice: Aunixt	158B - 10	Python 3	Accepted	280 ms	3240 KB	2024-10-17 10:47:25	2024-10-17 10:47:25		Compare

→ Source Copy

```
import math
n = int(input())
ls = list(map(int, input().split()))
a, b, c, d = 0, 0, 0, 0
car_num = 0
for i in ls:
    if i == 1:
        a += 1
    elif i == 2:
        b += 1
    elif i == 3:
        c += 1
    elif i == 4:
        d += 1

if c >= a:
    car_num = d + math.ceil(b/2) + c
else:
    if b % 2 == 0:
        car_num = d + c + b//2 +math.ceil((a-c)/4)
    else:
        car_num = d + c + math.ceil(b/2) + math.ceil((a-c-2)/4)
print(car_num)
```

*230B. T-primes（选做）

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

思路：

一个数是T-Primes，那么只有一个质因子且次数为二，即n为质数的平方。利用欧式筛找出小于1000000的质数，储存在集合中，如果n为完全平方数，且开根号后的质数在pr集合中，则输出YES（欧式筛理解了好久）

代码

```
import math

n = 1000000
pr = []
nprime = [True]*n
for i in range(2,n+1):
    if nprime[i-1]:
```

```
            pr.append(i)
        for p in pr:
            if p*i > n:
                break
            nprime[p*i-1] = False
            if i % p == 0:
                break
pr = set(pr)
n = int(input())
ls = list(map(int,input().split()))
for i in ls:
    x = math.sqrt(i)
    if i <= 3:
        print('NO')
        continue
    elif x % 1 != 0:
        print('NO')
        continue
    else:
       if x in pr:
           print('YES')
       else:
           print('NO')
```

代码运行截图 （至少包含有"Accepted"）

*12559: 最大最小整数 （选做）

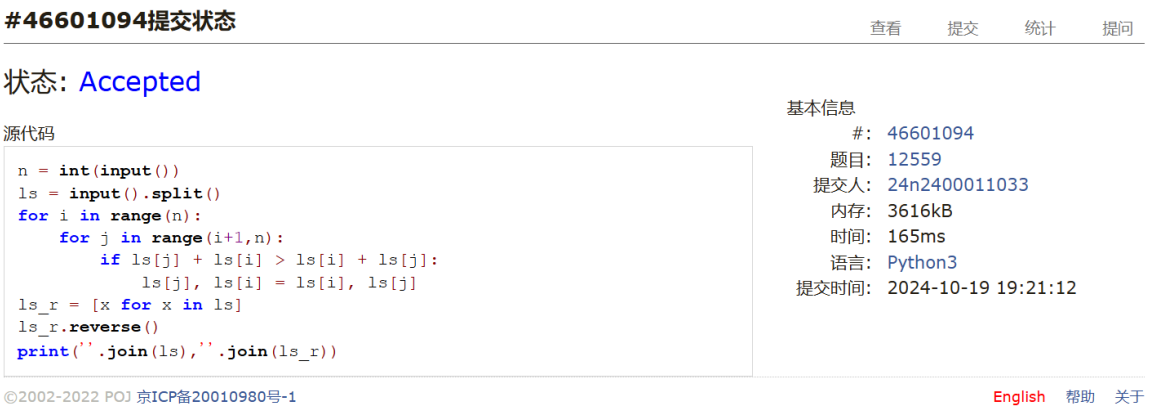
greedy, strings, sortings, http://cs101.openjudge.cn/practice/12559

思路：

代码

```python
n = int(input())
ls = input().split()
for i in range(n):
    for j in range(i+1,n):
        if ls[j] + ls[i] > ls[i] + ls[j]:
            ls[j], ls[i] = ls[i], ls[j]
ls_r = [x for x in ls]
ls_r.reverse()
print(''.join(ls),''.join(ls_r))
```

代码运行截图（至少包含有"Accepted"）



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。

前面几题都比较简单。看到taxi题目的题解答案这么简单被震撼到了。

关于T-Primes，因为之前就听说过要用埃氏筛或者欧式筛来解，于是提前学习了这种算法，非常巧妙

还学到了用in之前要把列表转换为集合，这样复杂度会降到O（1），提升效率

最大最小数原本在想一些其他的做法，后来发现一直WA，最后试了试最普通的一个个比较的方法发现直接AC了。

看了题解，学会了lambda

还是需要多学学经典的算法，多看看各位大佬优秀的算法