Assignment #4: T-primes + 贪心

Updated 0337 GMT+8 Oct 15, 2024

2024 fall, Complied by 任宇桐 物理学院

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）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

用时<10min

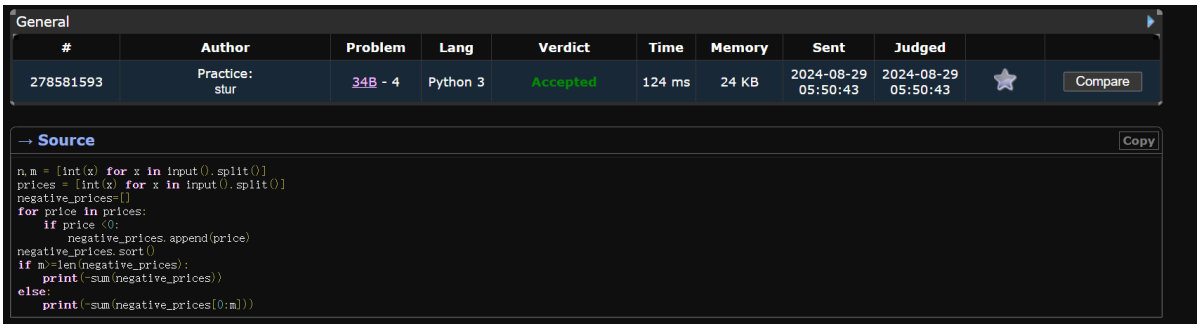
思路：

获得最多的钱，只需要赚钱即可

代码

```
n,m = [int(x) for x in input().split()]
prices = [int(x) for x in input().split()]
negative_prices=[]
for price in prices:
    if price <0:
        negative_prices.append(price)
negative_prices.sort()
if m>=len(negative_prices):
    print(-sum(negative_prices))
else:
    print(-sum(negative_prices[0:m]))
```

代码运行截图 （至少包含有"Accepted"）



160A. Twins

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

用时10min

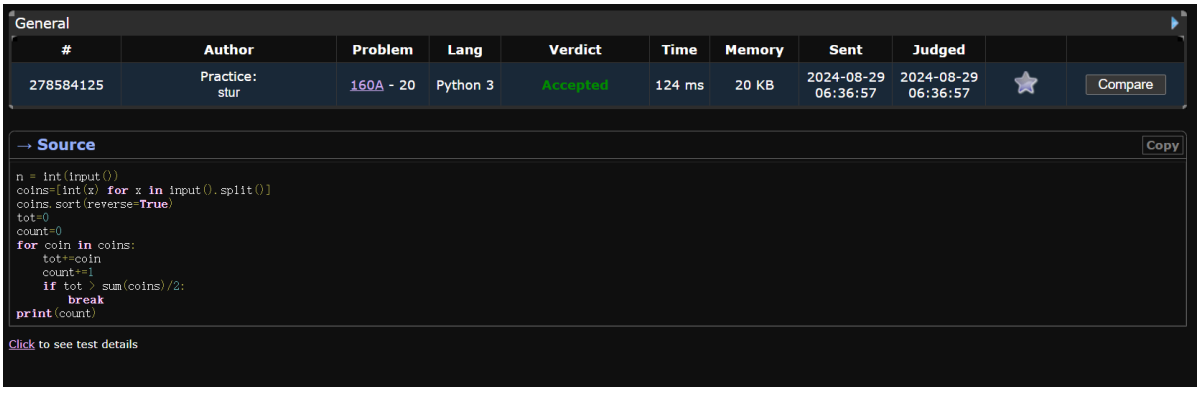
思路：

排序，每次数目即可

代码

```
n = int(input())
coins=[int(x) for x in input().split()]
coins.sort(reverse=True)
tot=0
count=0
for coin in coins:
    tot+=coin
    count+=1
    if tot > sum(coins)/2:
        break
print(count)
```

代码运行截图 （至少包含有"Accepted"）



1879B. Chips on the Board

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

用时>20min

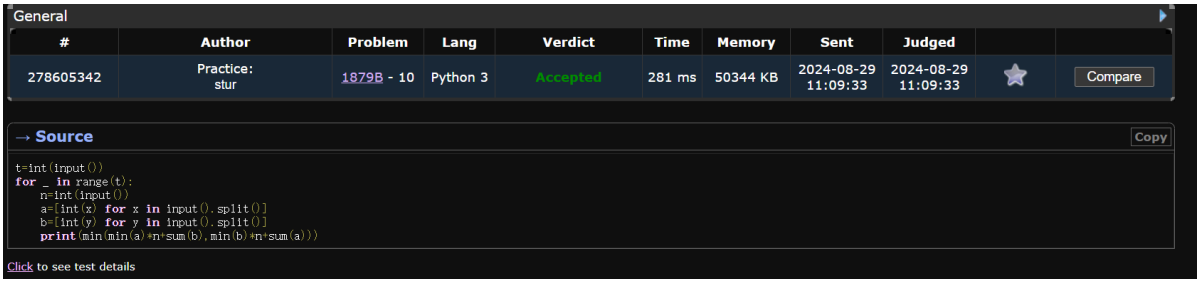
思路：

第一次做的时候思考了很久，然后突然发现可以排序计算。

代码

```
t=int(input())
for _ in range(t):
    n=int(input())
    a=[int(x) for x in input().split()]
    b=[int(y) for y in input().split()]
    print(min(min(a)*n+sum(b),min(b)*n+sum(a)))
```

代码运行截图（至少包含有"Accepted"）



158B. Taxi

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

用时10-20min

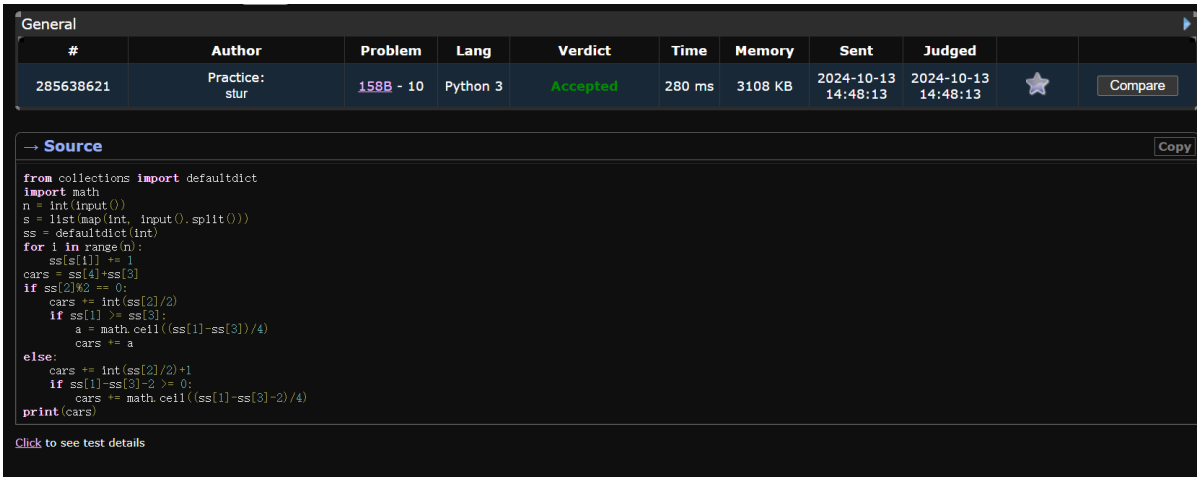
思路：

和装箱子问题十分相似，其实可以尽量减少反复if...else...的结构

代码

```
from collections import defaultdict
import math
n = int(input())
s = list(map(int, input().split()))
ss = defaultdict(int)
for i in range(n):
    ss[s[i]] += 1
cars = ss[4]+ss[3]
if ss[2]%2 == 0:
    cars += int(ss[2]/2)
    if ss[1] >= ss[3]:
        a = math.ceil((ss[1]-ss[3])/4)
        cars += a
else:
    cars += int(ss[2]/2)+1
    if ss[1]-ss[3]-2 >= 0:
        cars += math.ceil((ss[1]-ss[3]-2)/4)
print(cars)
```

代码运行截图（至少包含有"Accepted"）



*230B. T-primes（选做）

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

用时>30min

思路：

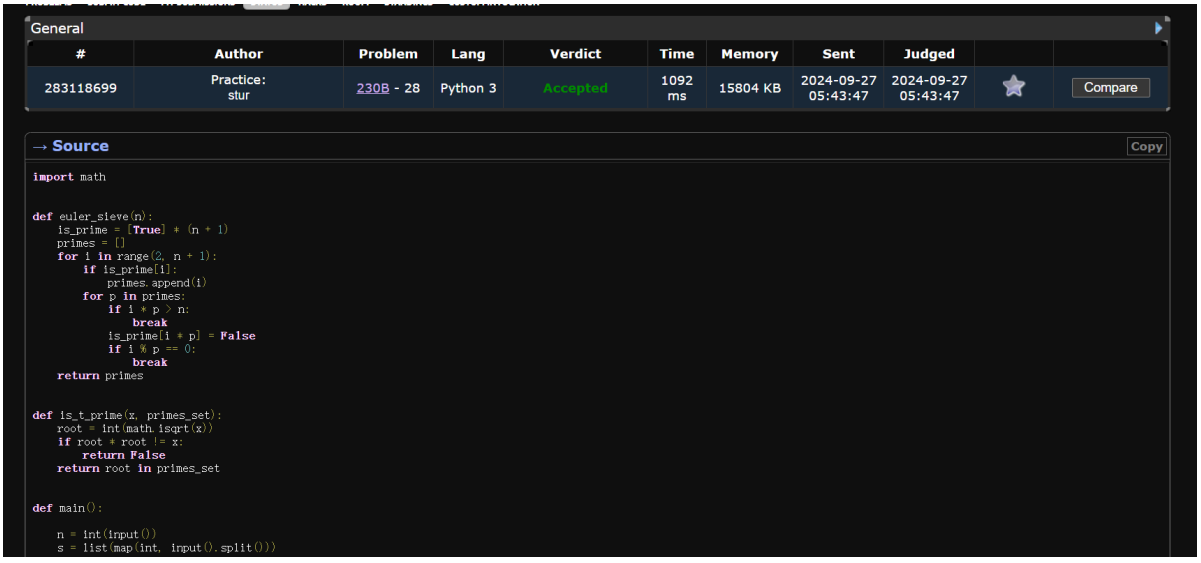
主要是不方便写出不超时判断素数的代码，后来学习了欧拉筛法，就成功搞定了。

代码

```
import math


def euler_sieve(n):
    is_prime = [True] * (n + 1)
    primes = []
    for i in range(2, n + 1):
        if is_prime[i]:
            primes.append(i)
        for p in primes:
            if i * p > n:
                break
            is_prime[i * p] = False
            if i % p == 0:
                break
    return primes


def is_t_prime(x, primes_set):
    root = int(math.isqrt(x))
    if root * root != x:
        return False
    return root in primes_set


def main():

    n = int(input())
```

```
    s = list(map(int, input().split()))

    max_num = 10 ** 6
    primes = euler_sieve(max_num)
    primes_set = set(primes)

    results = []
    for num in s:
        if is_t_prime(num, primes_set):
            results.append("YES")
        else:
            results.append("NO")

    for result in results:
        print(result)


if __name__ == "__main__":
    main()
```

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

用时<10min

思路：

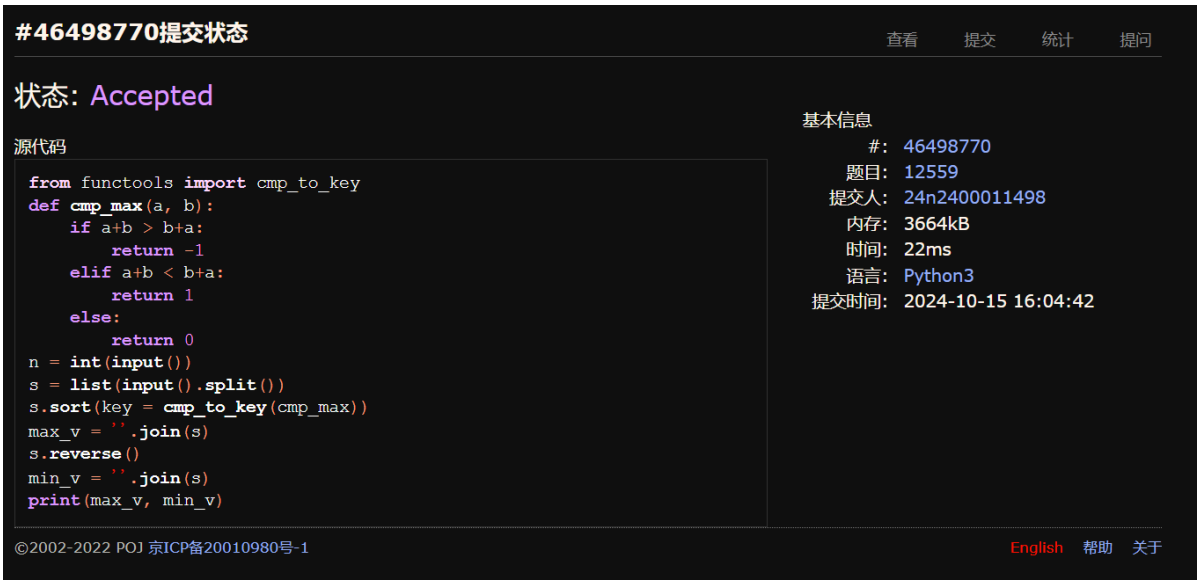
思路来自于晴问算法中的最大最小比较的问题，用了内置函数，时间复杂度有所降低，比较快速的解决了。

代码

```
from functools import cmp_to_key
def cmp_max(a, b):
    if a+b > b+a:
```

```
        return -1
    elif a+b < b+a:
        return 1
    else:
        return 0
n = int(input())
s = list(input().split())
s.sort(key = cmp_to_key(cmp_max))
max_v = ''.join(s)
s.reverse()
min_v = ''.join(s)
print(max_v, min_v)
```

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。

跟着每日选做开始练习排序、贪心等等算法，虽然做的挺痛苦，但是学会了思路以后解决类似的题目就简单多了，现在感觉逐渐进入状态了。