Assignment #D: Dec 月考

Updated 2044 GMT+8 Dec 10, 2023

2023 fall, Complied by 钟明衡 物理学院

说明：

1）Dec 月考： AC3 。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

操作系统：Windows_NT x64 10.0.19045

Python编程环境：Visual Studio Code 1.76.1

C/C++编程环境：Visual Studio Code 1.76.1

1. 题目

如果耗时太长，直接看解题思路，或者源码

18176: 2050年成绩计算

http://cs101.openjudge.cn/practice/18176/

思路：

打表把10000以内的质数都筛出来就可以。不过这题时间卡的比较死，一开始写的质数筛超时了，于是打表暴力破解

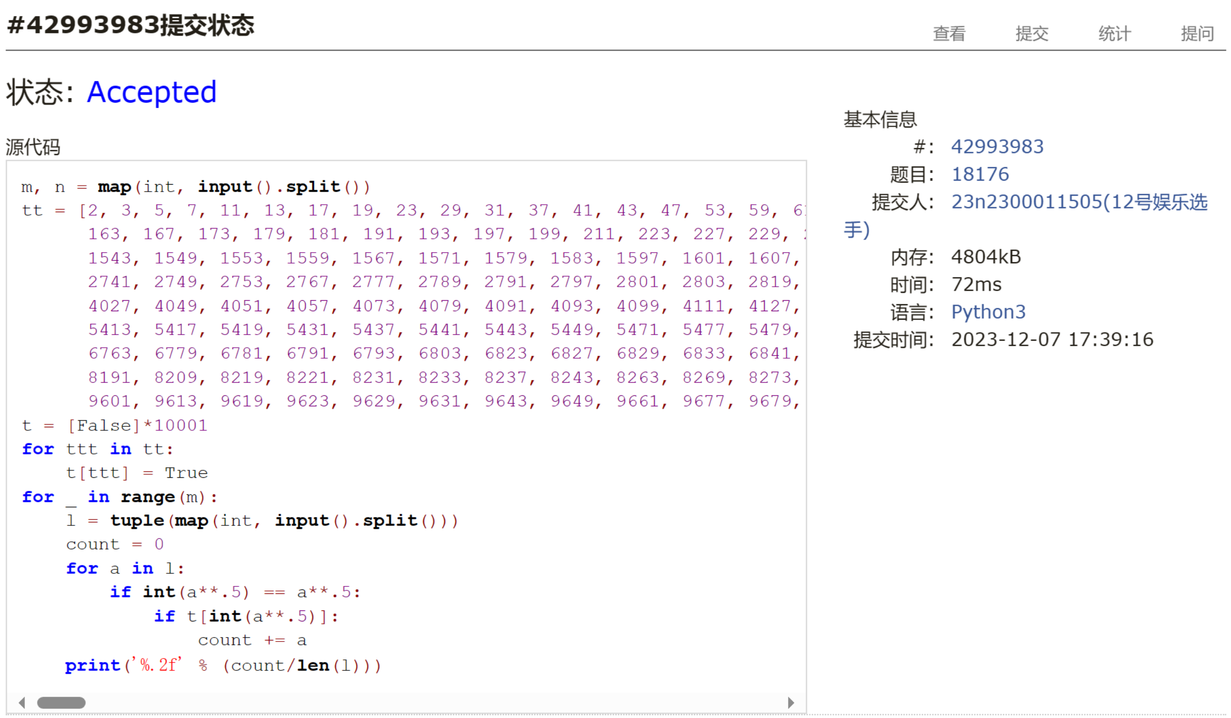
代码

```python
m, n = map(int, input().split())
```

```
tt = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67,
71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151,
157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239,
241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337,
347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433,
439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521, 523, 541,
547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641,
643, 647, 653, 659, 661, 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743,
751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827, 829, 839, 853, 857,
859, 863, 877, 881, 883, 887, 907, 911, 919, 929, 937, 941, 947, 953, 967, 971,
977, 983, 991, 997, 1009, 1013, 1019, 1021, 1031, 1033, 1039, 1049, 1051, 1061,
1063, 1069, 1087, 1091, 1093, 1097, 1103, 1109, 1117, 1123, 1129, 1151, 1153,
1163, 1171, 1181, 1187, 1193, 1201, 1213, 1217, 1223, 1229, 1231, 1237, 1249,
1259, 1277, 1279, 1283, 1289, 1291, 1297, 1301, 1303, 1307, 1319, 1321, 1327,
1361, 1367, 1373, 1381, 1399, 1409, 1423, 1427, 1429, 1433, 1439, 1447, 1451,
1453, 1459, 1471, 1481, 1483, 1487, 1489, 1493, 1499, 1511, 1523, 1531, 1543,
1549, 1553, 1559, 1567, 1571, 1579, 1583, 1597, 1601, 1607, 1609, 1613, 1619,
1621, 1627, 1637, 1657, 1663, 1667, 1669, 1693, 1697, 1699, 1709, 1721, 1723,
1733, 1741, 1747, 1753, 1759, 1777, 1783, 1787, 1789, 1801, 1811, 1823, 1831,
1847, 1861, 1867, 1871, 1873, 1877, 1879, 1889, 1901, 1907, 1913, 1931, 1933,
1949, 1951, 1973, 1979, 1987, 1993, 1997, 1999, 2003, 2011, 2017, 2027, 2029,
2039, 2053, 2063, 2069, 2081, 2083, 2087, 2089, 2099, 2111, 2113, 2129, 2131,
2137, 2141, 2143, 2153, 2161, 2179, 2203, 2207, 2213, 2221, 2237, 2239, 2243,
2251, 2267, 2269, 2273, 2281, 2287, 2293, 2297, 2309, 2311, 2333, 2339, 2341,
2347, 2351, 2357, 2371, 2377, 2381, 2383, 2389, 2393, 2399, 2411, 2417, 2423,
2437, 2441, 2447, 2459, 2467, 2473, 2477, 2503, 2521, 2531, 2539, 2543, 2549,
2551, 2557, 2579, 2591, 2593, 2609, 2617, 2621, 2633, 2647, 2657, 2659, 2663,
2671, 2677, 2683, 2687, 2689, 2693, 2699, 2707, 2711, 2713, 2719, 2729, 2731,
2741, 2749, 2753, 2767, 2777, 2789, 2791, 2797, 2801, 2803, 2819, 2833, 2837,
2843, 2851, 2857, 2861, 2879, 2887, 2897, 2903, 2909, 2917, 2927, 2939, 2953,
2957, 2963, 2969, 2971, 2999, 3001, 3011, 3019, 3023, 3037, 3041, 3049, 3061,
3067, 3079, 3083, 3089, 3109, 3119, 3121, 3137, 3163, 3167, 3169, 3181, 3187,
3191, 3203, 3209, 3217, 3221, 3229, 3251, 3253, 3257, 3259, 3271, 3299, 3301,
3307, 3313, 3319, 3323, 3329, 3331, 3343, 3347, 3359, 3361, 3371, 3373, 3389,
3391, 3407, 3413, 3433, 3449, 3457, 3461, 3463, 3467, 3469, 3491, 3499, 3511,
3517, 3527, 3529, 3533, 3539, 3541, 3547, 3557, 3559, 3571, 3581, 3583, 3593,
3607, 3613, 3617, 3623, 3631, 3637, 3643, 3659, 3671, 3673, 3677, 3691, 3697,
3701, 3709, 3719, 3727, 3733, 3739, 3761, 3767, 3769, 3779, 3793, 3797, 3803,
3821, 3823, 3833, 3847, 3851, 3853, 3863, 3877, 3881, 3889, 3907, 3911, 3917,
3919, 3923, 3929, 3931, 3943, 3947, 3967, 3989, 4001, 4003, 4007, 4013, 4019,
4021, 4027, 4049, 4051, 4057, 4073, 4079, 4091, 4093, 4099, 4111, 4127, 4129,
4133, 4139, 4153, 4157, 4159, 4177, 4201, 4211, 4217, 4219, 4229, 4231, 4241,
4243, 4253, 4259, 4261, 4271, 4273, 4283, 4289, 4297, 4327, 4337, 4339, 4349,
4357, 4363, 4373, 4391, 4397, 4409, 4421, 4423, 4441, 4447, 4451, 4457, 4463,
4481, 4483, 4493, 4507, 4513, 4517, 4519, 4523, 4547, 4549, 4561, 4567, 4583,
4591, 4597, 4603, 4621, 4637, 4639, 4643, 4649, 4651, 4657, 4663, 4673, 4679,
4691, 4703, 4721, 4723, 4729, 4733, 4751, 4759, 4783, 4787, 4789, 4793, 4799,
4801, 4813, 4817, 4831, 4861, 4871, 4877, 4889, 4903, 4909, 4919, 4931, 4933,
4937, 4943, 4951, 4957, 4967, 4969, 4973, 4987, 4993, 4999, 5003, 5009, 5011,
5021, 5023, 5039, 5051, 5059, 5077, 5081, 5087, 5099, 5101, 5107, 5113, 5119,
5147, 5153, 5167, 5171, 5179, 5189, 5197, 5209, 5227, 5231, 5233, 5237, 5261,
5273, 5279, 5281, 5297, 5303, 5309, 5323, 5333, 5347, 5351, 5381, 5387, 5393,
5399, 5407, 5413, 5417, 5419, 5431, 5437, 5441, 5443, 5449, 5471, 5477, 5479,
5483, 5501, 5503, 5507, 5519, 5521, 5527, 5531, 5557, 5563, 5569, 5573, 5581,
5591, 5623, 5639, 5641, 5647, 5651, 5653, 5657, 5659, 5669, 5683, 5689, 5693,
5701, 5711, 5717, 5737, 5741, 5743, 5749, 5779, 5783, 5791, 5801, 5807, 5813,
5821, 5827, 5839, 5843, 5849, 5851, 5857, 5861, 5867, 5869, 5879, 5881, 5897,
5903, 5923, 5927, 5939, 5953, 5981, 5987, 6007, 6011, 6029, 6037, 6043, 6047,
6053, 6067, 6073, 6079, 6089, 6091, 6101, 6113, 6121, 6131, 6133, 6143, 6151,
6163, 6173, 6197, 6199, 6203, 6211, 6217, 6221, 6229, 6247, 6257, 6263, 6269,
6271, 6277, 6287, 6299, 6301, 6311, 6317, 6323, 6329, 6337, 6343, 6353, 6359,
6361, 6367, 6373, 6379, 6389, 6397, 6421, 6427, 6449, 6451, 6469, 6473, 6481,
6491, 6521, 6529, 6547, 6551, 6553, 6563, 6569, 6571, 6577, 6581, 6599, 6607,
6619, 6637, 6653, 6659, 6661, 6673, 6679, 6689, 6691, 6701, 6703, 6709, 6719,
6733, 6737, 6761, 6763, 6779, 6781, 6791, 6793, 6803, 6823, 6827, 6829, 6833,
6841, 6857, 6863, 6869, 6871, 6883, 6899, 6907, 6911, 6917, 6947, 6949, 6959,
6961, 6967, 6971, 6977, 6983, 6991, 6997, 7001, 7013, 7019, 7027, 7039, 7043,
7057, 7069, 7079, 7103, 7109, 7121, 7127, 7129, 7151, 7159, 7177, 7187, 7193,
7207, 7211, 7213, 7219, 7229, 7237, 7243, 7247, 7253, 7283, 7297, 7307, 7309,
7321, 7331, 7333, 7349, 7351, 7369, 7393, 7411, 7417, 7433, 7451, 7457, 7459,
7477, 7481, 7487, 7489, 7499, 7507, 7517, 7523, 7529, 7537, 7541, 7547, 7549,
7559, 7561, 7573, 7577, 7583, 7589, 7591, 7603, 7607, 7621, 7639, 7643, 7649,
7669, 7673, 7681, 7687, 7691, 7699, 7703, 7717, 7723, 7727, 7741, 7753, 7757,
7759, 7789, 7793, 7817, 7823, 7829, 7841, 7853, 7867, 7873, 7877, 7879, 7883,
7901, 7907, 7919, 7927, 7933, 7937, 7949, 7951, 7963, 7993, 8009, 8011, 8017,
8039, 8053, 8059, 8069, 8081, 8087, 8089, 8093, 8101, 8111, 8117, 8123, 8147,
8161, 8167, 8171, 8179, 8191, 8209, 8219, 8221, 8231, 8233, 8237, 8243, 8263,
8269, 8273, 8287, 8291, 8293, 8297, 8311, 8317, 8329, 8353, 8363, 8369, 8377,
8387, 8389, 8419, 8423, 8429, 8431, 8443, 8447, 8461, 8467, 8501, 8513, 8521,
8527, 8537, 8539, 8543, 8563, 8573, 8581, 8597, 8599, 8609, 8623, 8627, 8629,
8641, 8647, 8663, 8669, 8677, 8681, 8689, 8693, 8699, 8707, 8713, 8719, 8731,
8737, 8741, 8747, 8753, 8761, 8779, 8783, 8803, 8807, 8819, 8821, 8831, 8837,
8839, 8849, 8861, 8863, 8867, 8887, 8893, 8923, 8929, 8933, 8941, 8951, 8963,
8969, 8971, 8999, 9001, 9007, 9011, 9013, 9029, 9041, 9043, 9049, 9059, 9067,
9091, 9103, 9109, 9127, 9133, 9137, 9151, 9157, 9161, 9173, 9181, 9187, 9199,
9203, 9209, 9221, 9227, 9239, 9241, 9257, 9277, 9281, 9283, 9293, 9311, 9319,
9323, 9337, 9341, 9343, 9349, 9371, 9377, 9391, 9397, 9403, 9413, 9419, 9421,
9431, 9433, 9437, 9439, 9461, 9463, 9467, 9473, 9479, 9491, 9497, 9511, 9521,
9533, 9539, 9547, 9551, 9587, 9601, 9613, 9619, 9623, 9629, 9631, 9643, 9649,
9661, 9677, 9679, 9689, 9697, 9719, 9721, 9733, 9739, 9743, 9749, 9767, 9769,
9781, 9787, 9791, 9803, 9811, 9817, 9829, 9833, 9839, 9851, 9857, 9859, 9871,
9883, 9887, 9901, 9907, 9923, 9929, 9931, 9941, 9949, 9967, 9973]
t = [False]*10001
for ttt in tt:
    t[ttt] = True
for _ in range(m):
    l = tuple(map(int, input().split()))
    count = 0
    for a in l:
        if int(a**.5) == a**.5:
            if t[int(a**.5)]:
                count += a
    print('%.2f' % (count/len(l)))

```

代码运行截图



#42993983提交状态

状态: Accepted

源代码

```
m, n = map(int, input().split())
tt = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 6
      163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229,
      1543, 1549, 1553, 1559, 1567, 1571, 1579, 1583, 1597, 1601, 1607,
      2741, 2749, 2753, 2767, 2777, 2789, 2791, 2797, 2801, 2803, 2819,
      4027, 4049, 4051, 4057, 4073, 4079, 4091, 4093, 4099, 4111, 4127,
      5413, 5417, 5419, 5431, 5437, 5441, 5443, 5449, 5471, 5477, 5479,
      6763, 6779, 6781, 6791, 6793, 6803, 6823, 6827, 6829, 6833, 6841,
      8191, 8209, 8219, 8221, 8231, 8233, 8237, 8243, 8263, 8269, 8273,
      9601, 9613, 9619, 9623, 9629, 9631, 9643, 9649, 9661, 9677, 9679,
t = [False]*10001
for ttt in tt:
    t[ttt] = True
for _ in range(m):
    l = tuple(map(int, input().split()))
    count = 0
    for a in l:
        if int(a**.5) == a**.5:
            if t[int(a**.5)]:
                count += a
    print('%.2f' % (count/len(l)))
```

基本信息
#: 42993983
题目: 18176
提交人: 23n2300011505(12号娱乐选手)
内存: 4804kB
时间: 72ms
语言: Python3
提交时间: 2023-12-07 17:39:16

18224: 找魔数

http://cs101.openjudge.cn/practice/18224

思路:

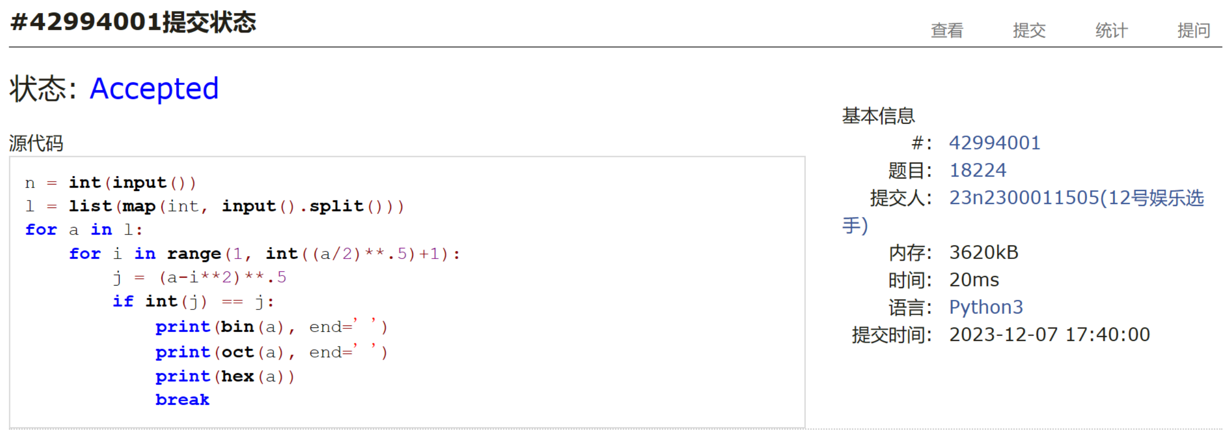
直接枚举即可。只需要枚举到平方根的一半，然后拿差判断是否为平方数

代码

```
n = int(input())
l = list(map(int, input().split()))
for a in l:
    for i in range(1, int((a/2)**.5)+1):
        j = (a-i**2)**.5
        if int(j) == j:
            print(bin(a), end=' ')
            print(oct(a), end=' ')
            print(hex(a))
            break

```

代码运行截图



#42994001提交状态

状态: Accepted

源代码

```
n = int(input())
l = list(map(int, input().split()))
for a in l:
    for i in range(1, int((a/2)**.5)+1):
        j = (a-i**2)**.5
        if int(j) == j:
            print(bin(a), end=' ')
            print(oct(a), end=' ')
            print(hex(a))
            break
```

基本信息

#: 42994001
题目: 18224
提交人: 23n2300011505(12号娱乐选手)
内存: 3620kB
时间: 20ms
语言: Python3
提交时间: 2023-12-07 17:40:00

19963: 买学区房

http://cs101.openjudge.cn/practice/19963

思路:

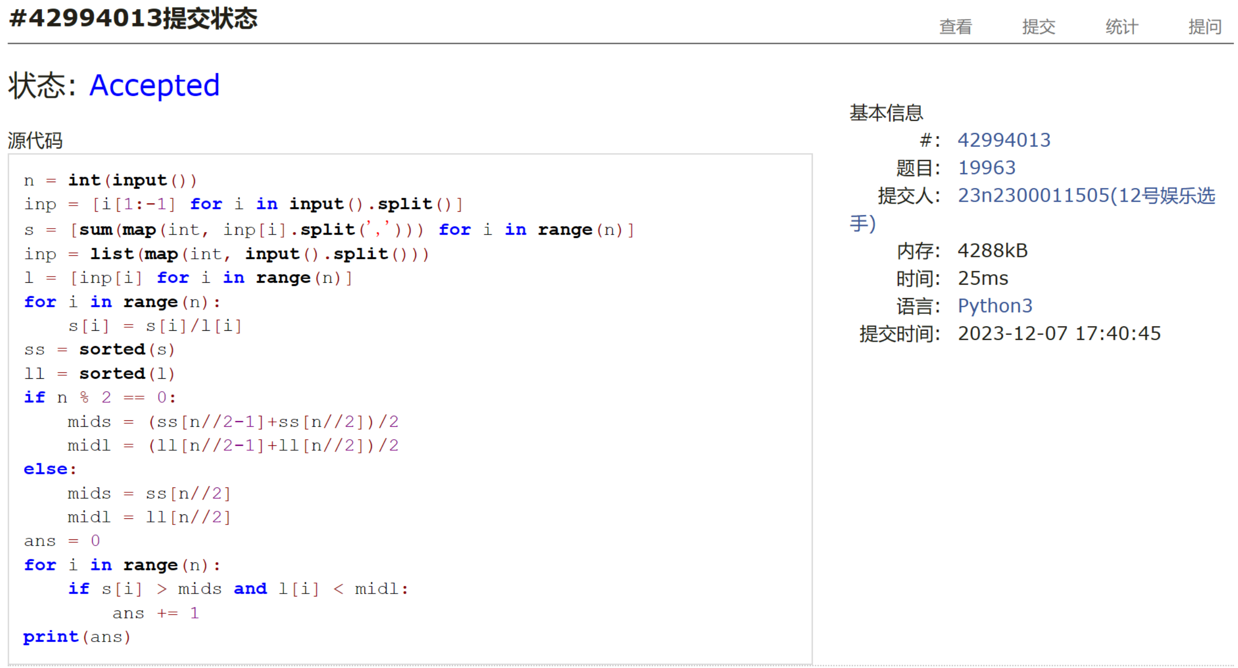
就按题目意思找到中位数就可以

代码

```python
n = int(input())
inp = [i[1:-1] for i in input().split()]
s = [sum(map(int, inp[i].split(','))) for i in range(n)]
inp = list(map(int, input().split()))
l = [inp[i] for i in range(n)]
for i in range(n):
    s[i] = s[i]/l[i]
ss = sorted(s)
ll = sorted(l)
if n % 2 == 0:
    mids = (ss[n//2-1]+ss[n//2])/2
    midl = (ll[n//2-1]+ll[n//2])/2
else:
    mids = ss[n//2]
    midl = ll[n//2]
ans = 0
for i in range(n):
    if s[i] > mids and l[i] < midl:
        ans += 1
print(ans)

```

代码运行截图



#42994013提交状态

状态: Accepted

源代码

```
n = int(input())
inp = [i[1:-1] for i in input().split()]
s = [sum(map(int, inp[i].split(','))) for i in range(n)]
inp = list(map(int, input().split()))
l = [inp[i] for i in range(n)]
for i in range(n):
    s[i] = s[i]/l[i]
ss = sorted(s)
ll = sorted(l)
if n % 2 == 0:
    mids = (ss[n//2-1]+ss[n//2])/2
    midl = (ll[n//2-1]+ll[n//2])/2
else:
    mids = ss[n//2]
    midl = ll[n//2]
ans = 0
for i in range(n):
    if s[i] > mids and l[i] < midl:
        ans += 1
print(ans)
```

基本信息

#: 42994013
题目: 19963
提交人: 23n2300011505(12号娱乐选手)
内存: 4288kB
时间: 25ms
语言: Python3
提交时间: 2023-12-07 17:40:45

23806: 三数之和

http://cs101.openjudge.cn/practice/23806/

思路:

也是枚举。三个数有以下四种情况:

1. 三个零

   当0的个数至少为3时会出现

2. 一正一负一零

   当0的个数至少为1时会出现

   直接把出现的正数枚举，然后判断是否在复数中出现过

3. 两负一正

   枚举一个正数和一个负数，负数到达绝对值一半时停止，判断剩下的负数是否存在

   注意当两个负数相等的情况，这时需要判断这个负数是否出现了2次

4. 两正一负

   同上

为了节约时间，列表c1中不重复记录数字，列表c2中记录重复了的数

由于数据在一开始进行了排序，可以采用二分查找来加快速度，判断一个数是否存在列表中。经过试验，方法是：如果`l[bisect.bisect(l,a)-1]==a`，那么a就在列表l中

注意判断空列表！我考试的时候就是没判断空列表，导致RE。一考完就立马改对了，损失惨重。

代码

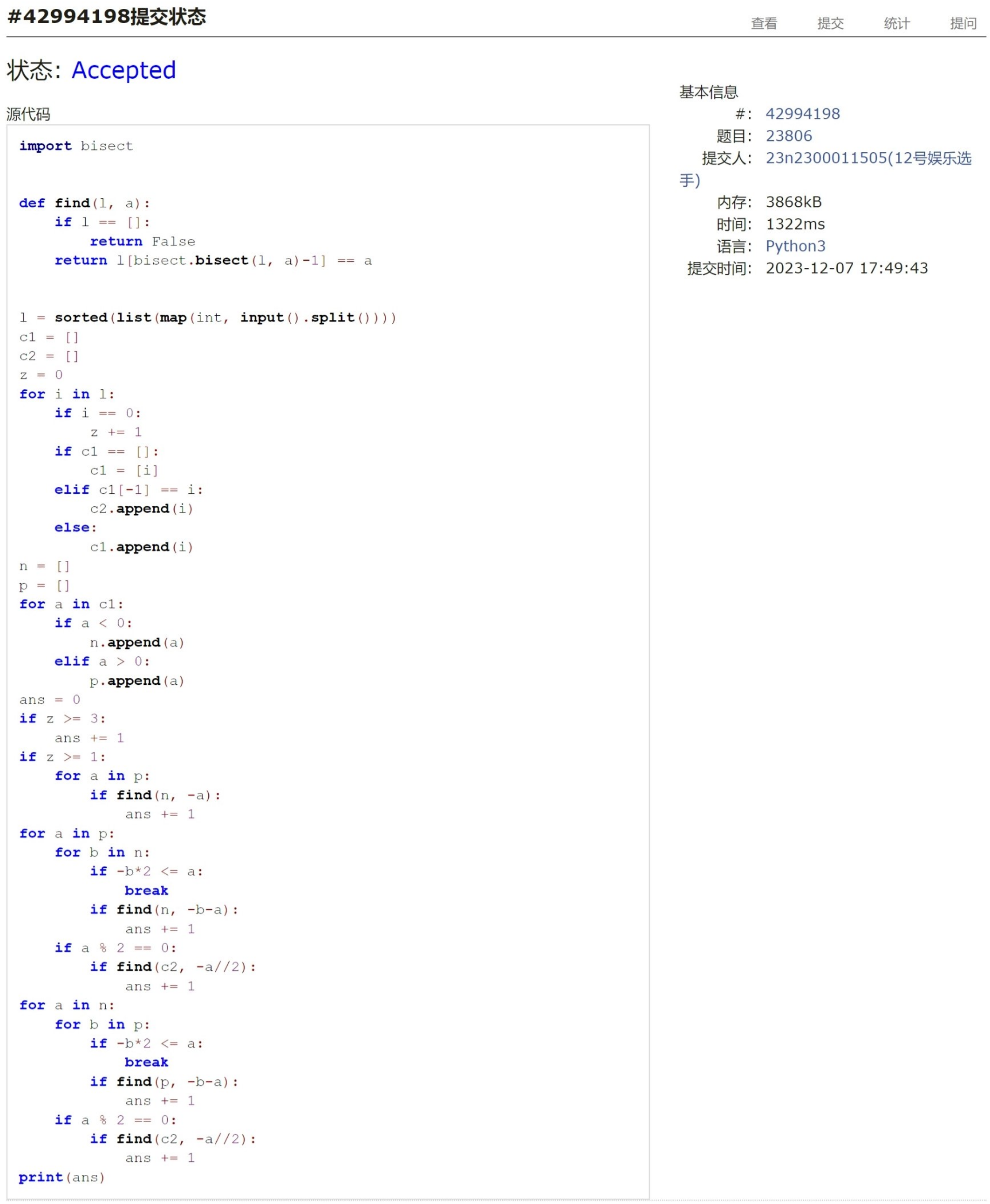
```python
import bisect


def find(l, a):
    if l == []:
        return False
    return l[bisect.bisect(l, a)-1] == a


l = sorted(list(map(int, input().split())))
c1 = []
c2 = []
z = 0
for i in l:
    if i == 0:
        z += 1
    if c1 == []:
        c1 = [i]
    elif c1[-1] == i:
        c2.append(i)
    else:
        c1.append(i)
n = []
p = []
for a in c1:
    if a < 0:
        n.append(a)
    elif a > 0:
        p.append(a)
ans = 0
if z >= 3:
    ans += 1
if z >= 1:
    for a in p:
        if find(n, -a):
            ans += 1
for a in p:
    for b in n:
        if -b*2 <= a:
            break
        if find(n, -b-a):
            ans += 1
    if a % 2 == 0:
        if find(c2, -a//2):
            ans += 1
for a in n:
    for b in p:
        if -b*2 <= a:
            break
        if find(p, -b-a):
            ans += 1
    if a % 2 == 0:
        if find(c2, -a//2):
            ans += 1
print(ans)

```

代码运行截图



#42994198提交状态

状态: Accepted

源代码

```python
import bisect


def find(l, a):
    if l == []:
        return False
    return l[bisect.bisect(l, a)-1] == a


l = sorted(list(map(int, input().split())))
c1 = []
c2 = []
z = 0
for i in l:
    if i == 0:
        z += 1
    if c1 == []:
        c1 = [i]
    elif c1[-1] == i:
        c2.append(i)
    else:
        c1.append(i)
n = []
p = []
for a in c1:
    if a < 0:
        n.append(a)
    elif a > 0:
        p.append(a)
ans = 0
if z >= 3:
    ans += 1
if z >= 1:
    for a in p:
        if find(n, -a):
            ans += 1
for a in p:
    for b in n:
        if -b*2 <= a:
            break
        if find(n, -b-a):
            ans += 1
    if a % 2 == 0:
        if find(c2, -a//2):
            ans += 1
for a in n:
    for b in p:
        if -b*2 <= a:
            break
        if find(p, -b-a):
            ans += 1
    if a % 2 == 0:
        if find(c2, -a//2):
            ans += 1
print(ans)
```

基本信息

#:	42994198
题目:	23806
提交人:	23n2300011505(12号娱乐选手)
内存:	3868kB
时间:	1322ms
语言:	Python3
提交时间:	2023-12-07 17:49:43

25561: 2022决战双十一

http://cs101.openjudge.cn/practice/25561/

思路：

看起来很复杂，但是实际上是个暴力枚举，只要处理好各量就没事。

思路就是每个商品在每个商店里面买，然后记录每个商店买的价格，最后算优惠（优先用大额的优惠券）就可以。

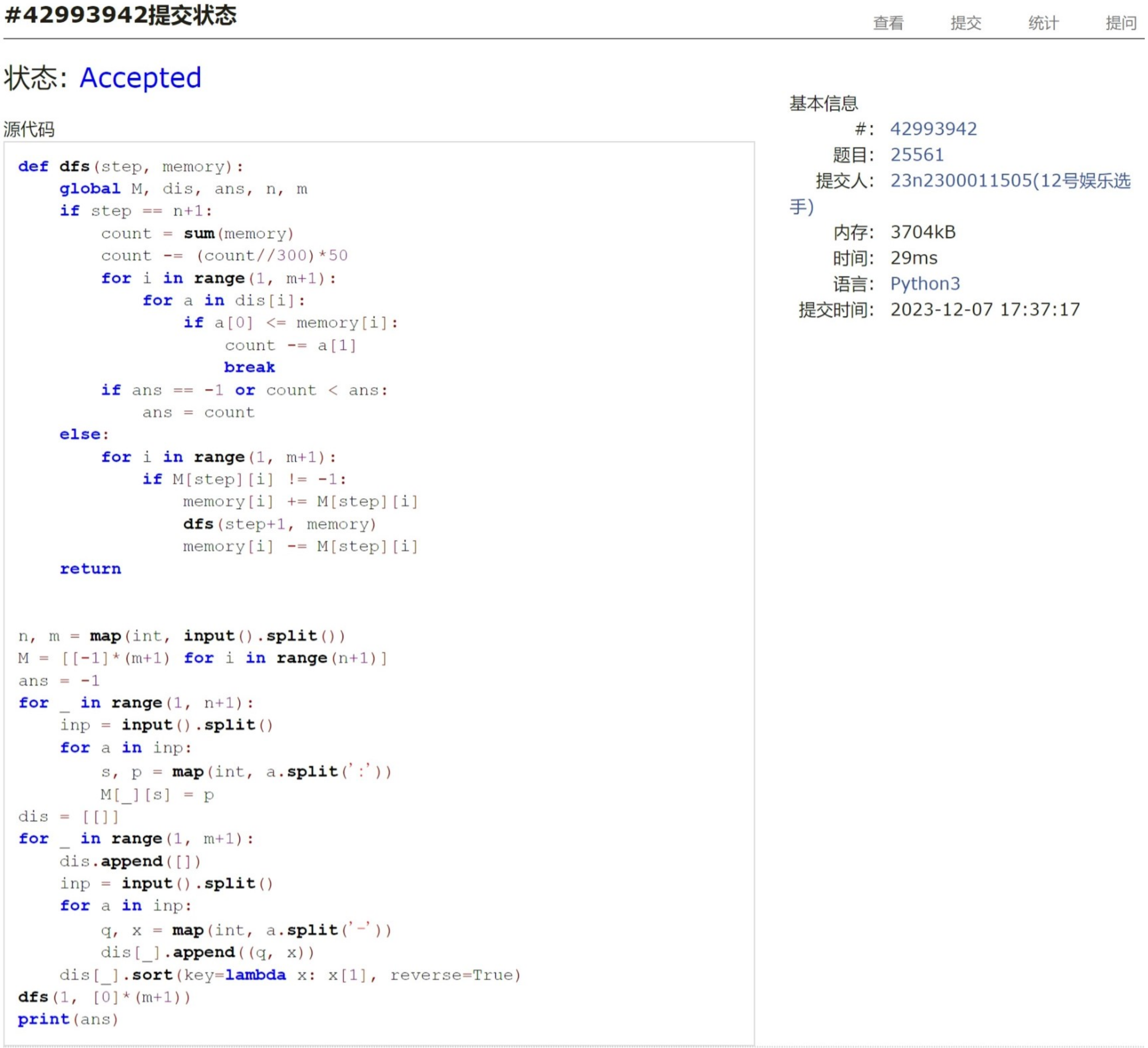
代码

```python
def dfs(step, memory):
    global M, dis, ans, n, m
    if step == n+1:
        count = sum(memory)
        count -= (count//300)*50
        for i in range(1, m+1):
            for a in dis[i]:
                if a[0] <= memory[i]:
                    count -= a[1]
                    break
        if ans == -1 or count < ans:
            ans = count
    else:
        for i in range(1, m+1):
            if M[step][i] != -1:
                memory[i] += M[step][i]
                dfs(step+1, memory)
                memory[i] -= M[step][i]
    return


n, m = map(int, input().split())
M = [[-1]*(m+1) for i in range(n+1)]
ans = -1
for _ in range(1, n+1):
    inp = input().split()
    for a in inp:
        s, p = map(int, a.split(':'))
        M[_][s] = p
dis = [[]]
for _ in range(1, m+1):
    dis.append([])
    inp = input().split()
    for a in inp:
        q, x = map(int, a.split('-'))
        dis[_].append((q, x))
    dis[_].sort(key=lambda x: x[1], reverse=True)
dfs(1, [0]*(m+1))
print(ans)

```

代码运行截图



#42993942提交状态

状态: Accepted

源代码

```
def dfs(step, memory):
    global M, dis, ans, n, m
    if step == n+1:
        count = sum(memory)
        count -= (count//300)*50
        for i in range(1, m+1):
            for a in dis[i]:
                if a[0] <= memory[i]:
                    count -= a[1]
                    break
        if ans == -1 or count < ans:
            ans = count
    else:
        for i in range(1, m+1):
            if M[step][i] != -1:
                memory[i] += M[step][i]
                dfs(step+1, memory)
                memory[i] -= M[step][i]
    return


n, m = map(int, input().split())
M = [[-1]*(m+1) for i in range(n+1)]
ans = -1
for _ in range(1, n+1):
    inp = input().split()
    for a in inp:
        s, p = map(int, a.split(':'))
        M[_][s] = p
dis = [[]]
for _ in range(1, m+1):
    dis.append([])
    inp = input().split()
    for a in inp:
        q, x = map(int, a.split('-'))
        dis[_].append((q, x))
    dis[_].sort(key=lambda x: x[1], reverse=True)
dfs(1, [0]*(m+1))
print(ans)
```

基本信息

#:	42993942
题目:	25561
提交人:	23n2300011505(12号娱乐选手)
内存:	3704kB
时间:	29ms
语言:	Python3
提交时间:	2023-12-07 17:37:17

08210: 河中跳房子

http://cs101.openjudge.cn/practice/08210/

思路：

直接模拟会超时，要换个思路。

每个最短距离都对应了一个取石头的次数，当m恰好满足，则那个最短距离就是答案。由于最短距离和m单调正相关，可以采用二分法。

求取石头的次数的方法是，从头到尾逐个排查，如果长度短于要求，就取掉一块石头，最后比较取掉的石头数和m

代码

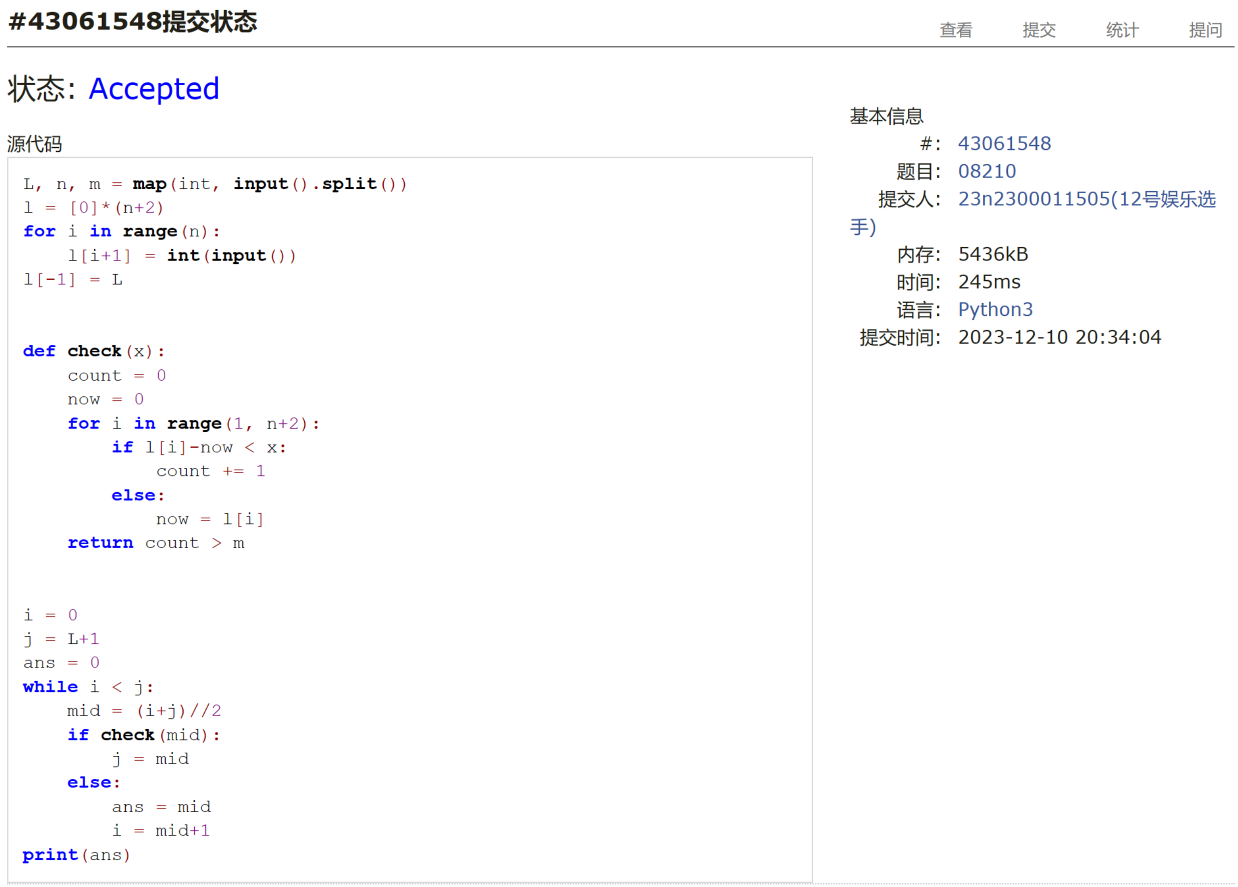
```
L, n, m = map(int, input().split())
l = [0]*(n+2)
for i in range(n):
    l[i+1] = int(input())
l[-1] = L


def check(x):
    count = 0
    now = 0
    for i in range(1, n+2):
        if l[i]-now < x:
            count += 1
        else:
            now = l[i]
    return count > m


i = 0
j = L+1
ans = 0
while i < j:
    mid = (i+j)//2
    if check(mid):
        j = mid
    else:
        ans = mid
        i = mid+1
print(ans)

```

代码运行截图



#43061548提交状态

状态: Accepted

源代码

```
L, n, m = map(int, input().split())
l = [0]*(n+2)
for i in range(n):
    l[i+1] = int(input())
l[-1] = L


def check(x):
    count = 0
    now = 0
    for i in range(1, n+2):
        if l[i]-now < x:
            count += 1
        else:
            now = l[i]
    return count > m


i = 0
j = L+1
ans = 0
while i < j:
    mid = (i+j)//2
    if check(mid):
        j = mid
    else:
        ans = mid
        i = mid+1
print(ans)
```

基本信息

#: 43061548
题目: 08210
提交人: 23n2300011505(12号娱乐选手)
内存: 5436kB
时间: 245ms
语言: Python3
提交时间: 2023-12-10 20:34:04

01922: Ride to School

http://cs101.openjudge.cn/practice/01922/

思路：

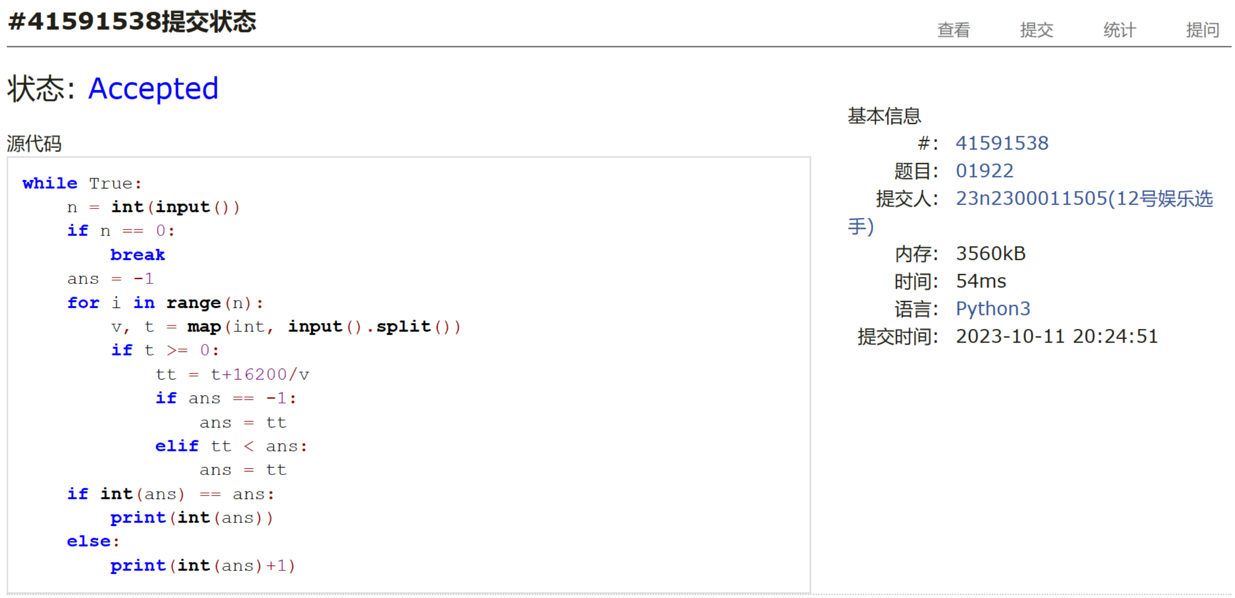
最终的答案是路上能遇到的车中最早到达的时间。只要发车时间不小于0，到达时间最早的车都能遇到，直接在出发时间大于等于0的车里面找出到达时间最早的即可。

代码

```
while True:
    n = int(input())
    if n == 0:
        break
    ans = -1
    for i in range(n):
        v, t = map(int, input().split())
        if t >= 0:
            tt = t+16200/v
            if ans == -1:
                ans = tt
            elif tt < ans:
                ans = tt
    if int(ans) == ans:
        print(int(ans))
    else:
        print(int(ans)+1)

```

代码运行截图



#41591538提交状态

状态: Accepted

源代码

```
while True:
    n = int(input())
    if n == 0:
        break
    ans = -1
    for i in range(n):
        v, t = map(int, input().split())
        if t >= 0:
            tt = t+16200/v
            if ans == -1:
                ans = tt
            elif tt < ans:
                ans = tt
    if int(ans) == ans:
        print(int(ans))
    else:
        print(int(ans)+1)
```

基本信息

#: 41591538

题目: 01922

提交人: 23n2300011505(12号娱乐选手)

内存: 3560kB

时间: 54ms

语言: Python3

提交时间: 2023-10-11 20:24:51

2. 学习总结和收获

本次月考有些难，而且感觉没发挥出真实水平。在第一题上耽误了很多时间才过，导致后面时间不足。

三数之和那道题，一开始TLE，改用二分以后RE了，考试期间一直没过。考完以后猛然想起没排除空列表情况，加了一个立马就AC了。

决战双十一考完试也很快写完了，如果第一题没有卡那么久，是可以当场写完的。

跳房子那题考试时完全没有思路，后来做了也超时，最后换成二分才做出。

快要期末考了，我这个水平还是说得过去，但考场上如果像这样发挥那肯定不行。还要多加练习。