

```
# Assignment #4: T-primes + 贪心
Updated 1814 GMT+8 Sep 30, 2025
2025 fall, Compiled by <mark>
同学的姓名、院系</mark>
```

郭旭杰、化学与分子工程学院

>**说明: **

>老师不好意思，本人的 Typora 已经过期，暂时无法使用，敬请谅解。

>1. **解题与记录: **

>

> 对于每一个题目，请提供其解题思路（可选），并附上使用 Python 或 C++ 编写的源代码（确保已在

OpenJudge，Codeforces，LeetCode 等平台上获得 Accepted ”）。请将这些信息连同显示 Accepted”的

截图一起填写到下方的作业模板中。（推荐使用 Typora

<https://typoraiocn> 进行编辑，当然你也可以选

择 Word。）无论题目是否已通过，请标明每个题目大致花费的时间。

>

>2. 提交安排: **提交时，请首先上传 PDF 格式的文件，并将 .md 或 .doc 格式的文件作为附件上传至右侧的

“作业评论区”。确保你的 Canvas 账户有一个清晰可见的本人头像，提交的文件为 PDF “格式，并且 作业评论区

包含上传的 .md 或 .doc 附件。

>

>4. **延迟提交: **如果你预计无法在截止日期前提交作业，请提前告知具体原因。这有助于我们了解情况并可能为你提供适当的延期或其他帮助。

>

>请按照上述指导认真准备和提交作业，以保证顺利完成课程要求。

1. 题目

34B. Sale

greedy, sorting, 900,

<https://codeforces.com/problemset/problem/34/B>

思路: 20 分钟通过

My Submissions								
#	When	Who	Problem	Lang	Verdict	Time	Memory	
340592265	Sep/26/2025 17:32 UTC+8	LittleBeetroot	B - Sale	Python 3	Accepted	156 ms	0 KB	
340591918	Sep/26/2025 17:29 UTC+8	LittleBeetroot	B - Sale	Python 3	Wrong answer on test 13	122 ms	0 KB	
340591724	Sep/26/2025 17:27 UTC+8	LittleBeetroot	B - Sale	Python 3	Runtime error on test 1	154 ms	2200 KB	
340591582	Sep/26/2025 17:25 UTC+8	LittleBeetroot	B - Sale	Python 3	Runtime error on test 1	124 ms	2200 KB	

代码

```
```python
```

```
`n,m = map(int,input().split())
tvs = list(map(int,input().split()))
wage = 0
pre_target = []
for i in tvs:
 if i < 0:
 pre_target.append(-i)
target = sorted(pre_target, reverse=True)
p = 0
while p < m and p < len(target):
 wage += int(target[p])
 p += 1
print(wage)
``
```

代码运行截图 <mark> (至少包含有"Accepted") </mark>

The screenshot shows a Codeforces submission page. At the top, there's a navigation bar with links like HOME, TOP, CATALOG, CONTESTS, GYM, PROBLEMSET, GROUPS, RATING, EDU, API, CALENDAR, HELP, and RAYAN. Below the navigation is a search bar and a user profile section with a bell icon, a flag, and the user name LittleBeetroot.

The main area displays the submission details:

General									
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged	
340592265	Practice: LittleBeetroot	34B - 4	Python 3	Accepted	156 ms	36 KB	2025-09-26 12:32:42	2025-09-26 12:32:43	Compare

Below the table is a "Source" code editor containing the provided Python code. A "Copy" button is located at the top right of the editor.

At the bottom of the page, there's a link "Click to see test details".

### 160A. Twins

greedy, sortings, 900,

<https://codeforces.com/problemset/problem/160/A>

思路: 20 分钟通过

The screenshot shows the "My Submissions" section of the user's profile. It lists three submissions for problem 160A:

#	When	Who	Problem	Lang	Verdict	Time	Memory
340593692	Sep/26/2025 17:46 UTC+8	LittleBeetroot	A - Twins	Python 3	Accepted	154 ms	1200 KB
340593523	Sep/26/2025 17:45 UTC+8	LittleBeetroot	A - Twins	Python 3	Wrong answer on test 1	122 ms	0 KB
340593360	Sep/26/2025 17:43 UTC+8	LittleBeetroot	A - Twins	Python 3	Wrong answer on test 1	122 ms	0 KB

代码

```
```python
```

```
n = int(input())
js = list(map(int, input().split()))
sgm = 0
earn = 0
for i in js:
    sgm += i
greed = (sgm+2) // 2
trick = sorted(js, reverse=True)
j = 0
while earn < greed:
    earn += int(trick[j])
    j += 1
print(j)
```

```
```
```

代码运行截图 <mark> (至少包含有"Accepted") </mark>

The screenshot shows a Codeforces contest page. The top navigation bar includes links for HOME, TOP, CATALOG, CONTESTS (which is underlined), GYM, PROBLEMSET, GROUPS, RATING, EDU, API, CALENDAR, HELP, and RAYAN. A search bar is also present. The main content area displays a table of submissions. One submission is highlighted, showing details: Problem 160A - 20, Author Practice: LittleBeetroot, Language Python 3, Verdict Accepted, Time 154 ms, Memory 1164 KB, Sent 2025-09-26 12:46:30, Judged 2025-09-26 12:46:31. Below the table, the source code is shown:

```
n = int(input())
js = list(map(int, input().split()))
sgm = 0
earn = 0
for i in js:
 sgm += i
greed = (sgm+2) // 2
trick = sorted(js, reverse=True)
j = 0
while earn < greed:
 earn += int(trick[j])
 j += 1
print(j)
```

### 1879B. Chips on the Board

constructive algorithms, greedy, 900,

<https://codeforces.com/problemset/problem/1879/B>

思路: **10分钟速通**

代码

```
```python
```

```
line = int(input())
```

```
for i in range(line):
    n=int(input())
    a=list(map(int, input().split()))
    b=list(map(int, input().split()))
    sgm_1 = sgm_2 = 0
    for j in range(n):
        sgm_1 += a[j]
        sgm_2 += b[j]
    sgm_1 += n * min(b)
    sgm_2 += n * min(a)
    if sgm_1 > sgm_2:
        print(sgm_2)
    else:
        print(sgm_1)
```

```

代码运行截图 <mark> (至少包含有"Accepted") </mark>

HOME TOP CATALOG CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP RAYAN 🇮🇳

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS STANDINGS CUSTOM INVOCATION

| General   |                             |            |          |          |        |          |                        |                        |         |
|-----------|-----------------------------|------------|----------|----------|--------|----------|------------------------|------------------------|---------|
| #         | Author                      | Problem    | Lang     | Verdict  | Time   | Memory   | Sent                   | Judged                 |         |
| 340595466 | Practice:<br>LittleBeetroot | 1879B - 10 | Python 3 | Accepted | 390 ms | 50744 KB | 2025-09-26<br>13:03:38 | 2025-09-26<br>13:03:40 | Compare |

→ Source Copy

```
line = int(input())
for i in range(line):
 n=int(input())
 a=list(map(int, input().split()))
 b=list(map(int, input().split()))
 sgn_1 = sgn_2 = 0
 for j in range(n):
 sgn_1 += a[j]
 sgn_2 += b[j]
 sgn_1 += n * min(b)
 sgn_2 += n * min(a)
 if sgn_1 > sgn_2:
 print(sgn_2)
 else:
 print(sgn_1)
```

[Click to see test details](#)

### ### M01017: 装箱问题

greedy, <http://cs101.openjudge.cn/pctbook/M01017/>

思路: 陆续做了三天

OpenJudge

题目ID, 标题, 描述 郭旭杰 信箱 账号 ▾

CS101 / 计算思维算法实践

题目 排名 状态 提问

**M01017:装箱问题**

总时间限制: 1000ms 内存限制: 65536kB

**描述**

一个工厂制造的产品形状都是长方体，它们的高度都是 $h$ ，长和宽都相等，一共有六个型号，他们的长宽分别为 $1*1, 2*2, 3*3, 4*4, 5*5, 6*6$ 。这些产品通常使用一个 $6*6*h$ 的长方体包裹包装然后邮寄给客户。因为邮费很贵，所以工厂要想方设法的减小每个订单运送时的包裹数量。他们很需要一个好的程序帮他们解决这个问题从而节省费用。现在这个程序由你来设计。

**输入**

输入文件包括几行，每一行代表一个订单。每个订单里的一行包括六个整数，中间用空格隔开，分别为 $1*1$ 至 $6*6$ 这六种产品的数量。输入文件将以6个0组成的一行结尾。

**输出**

除了输入的最后一行6个0以外，输入文件里每一行对应着输出文件的一行，每一行输出一个整数代表对应的订单所需的小包裹数。

**样例输入**

```
0 0 4 0 0 1
7 5 1 0 0 0
0 0 0 0 0 0
```

**样例输出**

```
2
1
```

**全局题号 19**

添加于 2025-03-13

提交次数 679

尝试人数 157

通过人数 148

你的提交记录

| # | 结果           | 时间         |
|---|--------------|------------|
| 9 | Accepted     | 2025-10-05 |
| 8 | Accepted     | 2025-10-05 |
| 7 | Wrong Answer | 2025-10-05 |
| 6 | Wrong Answer | 2025-10-04 |
| 5 | Wrong Answer | 2025-10-04 |
| 4 | Wrong Answer | 2025-10-04 |
| 3 | Wrong Answer | 2025-10-04 |
| 2 | Wrong Answer | 2025-10-03 |
| 1 | Wrong Answer | 2025-10-03 |

### 来源

Central Europe 1996

代码

```python

```
pkg = 1
u = {0:0,1:5,2:3,3:1}
while pkg > 0:
    a,b,c,d,e,f = list(map(int, input().split()))
    pkg = f + e + d + (c + 3) // 4
    y = 5*d + u[c % 4]
    if b > y:
        pkg += (b - y + 8) // 9
    x = 36 * pkg - 36 * f - 25 * e - 16 * d - 9 * c - 4 * b
    if a > x:
        pkg += (a - x + 35) // 36
    if pkg > 0:
        print(pkg)
```

#

```

代码运行截图 <mark> (至少包含有"Accepted") </mark>

OpenJudge 题目ID, 标题, 描述 郭旭杰 信箱 账号

CS101 / 计算思维算法实践 题目 排名 状态 提问

#50230931提交状态 查看 提交 统计 提问

状态: Accepted

基本信息  
#: 50230931  
题目: M01017  
提交人: 郭旭杰  
内存: 3660kB  
时间: 32ms  
语言: Python3  
提交时间: 2025-10-05 13:36:14

源代码

```
pkg = 1
u = {0:0,1:5,2:3,3:1}
while pkg > 0:
 a,b,c,d,e,f = list(map(int, input().split()))
 pkg = f + e + d + (c + 3) // 4
 y = 5*d + u[c % 4]
 if b > y:
 pkg += (b - y + 8) // 9
 x = 36 * pkg - 36 * f - 25 * e - 16 * d - 9 * c - 4 * b
 if a > x:
 pkg += (a - x + 35) // 36
 if pkg > 0:
 print(pkg)
```

©2002-2022 POJ 京ICP备20010980号-1 English 帮助 关于

### M01008: Maya Calendar

implementation, <http://cs101.openjudge.cn/practice/01008/>

思路: 做了一个上午的时间

OpenJudge

题目ID, 标题, 描述  郭旭杰 信箱 账号 -

**CS101 / 题库 (包括计概、数算题目)**

题目 排名 状态 提问

## 01008:Maya Calendar

总时间限制: 1000ms 内存限制: 65536kB

### 描述

During his last sabbatical, professor M. A. Ya made a surprising discovery about the old Maya calendar. From an old knotted message, professor discovered that the Maya civilization used a 365 day long year, called Haab, which had 19 months. Each of the first 18 months was 20 days long, and the names of the months were pop, no, zip, zotz, tzec, xul, yoxkin, mol, chen, yax, zac, ceh, mac, kankin, muan, pax, koyab, cumhu. Instead of having names, the days of the months were denoted by numbers starting from 0 to 19. The last month of Haab was called uayet and had 5 days denoted by numbers 0, 1, 2, 3, 4. The Maya believed that this month was unlucky, the court of justice was not in session, the trade stopped, people did not even sweep the floor. For religious purposes, the Maya used another calendar in which the year was called Tzolkin (holly year). The year was divided into thirteen periods, each 20 days long. Each day was denoted by a pair consisting of a number and the name of the day. They used 20 names: imix, ik, akbal, kan, chicchan, cimi, manik, lamat, muluk, ok, chuen, eb, ben, ix, mem, cib, caban, eznab, canac, ahau and 13 numbers; both in cycles. Notice that each day has an unambiguous description. For example, at the beginning of the year the days were described as follows: 1 imix, 2 ik, 3 akbal, 4 kan, 5 chicchan, 6 cimi, 7 manik, 8 lamat, 9 muluk, 10 ok, 11 chuen, 12 eb, 13 ben, 1 ix, 2 mem, 3 cib, 4 caban, 5 eznab, 6 canac, 7 ahau, and again in the next period 8 imix, 9 ik, 10 akbal . . . Years (both Haab and Tzolkin) were denoted by numbers 0, 1, . . . , where the number 0 was the beginning of the world. Thus, the first day was: Haab: 0. pop 0 Tzolkin: 1 imix 0 Help professor M. A. Ya and write a program for him to convert the dates from the Haab calendar to the Tzolkin calendar.

### 输入

The date in Haab is given in the following format:  
NumberOfTheDay. Month Year

The first line of the input file contains the number of the input dates in the file. The next n lines contain n dates in the Haab calendar format, each in separate line. The year is smaller than 5000.

### 代码

```
```python
```

```
n = int(input())
print(n)
for i in range(n):
    a,b,c=list(map(str,input().split()))

Haab={"pop":1,"no":2,"zip":3,"zotz":4,"tzec":5,"xul":6,"yoxkin":7
,"mol":8,"chen":9,"yax":10,"zac":11,"ceh":12,"mac":13,"kankin":14
,"muan":15,"pax":16,"koyab":17,"cumhu":18,"uayet":19}

Tzolkin={1:"imix",2:"ik",3:"akbal",4:"kan",5:"chicchan",6:"cimi",
7:"manik",8:"lamat",9:"muluk",10:"ok",11:"chuen",12:"eb",13:"ben"
,14:"ix",15:"mem",16:"cib",17:"caban",18:"eznab",19:"canac",0:"ah
au"}
```

查看 提交 统计 提问

全局题号 **10**
添加于 **2023-09-18**
提交次数 **1447**
尝试人数 **332**
通过人数 **324**

你的提交记录

#	结果	时间
17	Accepted	2025-10-05
16	Accepted	2025-10-05
15	Wrong Answer	2025-10-05
14	Wrong Answer	2025-10-05
13	Wrong Answer	2025-10-05
12	Wrong Answer	2025-10-05
11	Wrong Answer	2025-10-05
10	Wrong Answer	2025-10-05
9	Runtime Error	2025-10-05
8	Runtime Error	2025-10-05
7	Runtime Error	2025-10-05
6	Runtime Error	2025-10-05
5	Runtime Error	2025-10-05
4	Runtime Error	2025-10-05
3	Runtime Error	2025-10-05
2	Runtime Error	2025-10-05
1	Runtime Error	2025-10-05

```

sgm = 20*(Haab[b] - 1) + _a + _c*365
y = sgm // 260
m = Tzolkin[(sgm+1) % 20]
d = sgm % 13 + 1
print(d,m,y)

```

```

代码运行截图 <mark> (至少包含有"Accepted") </mark>

The screenshot shows the OpenJudge interface. At the top, there's a search bar with placeholder text "题目ID, 标题, 描述" and a search icon. To the right are links for "郭旭杰", "信箱", and "账号". Below the search bar is a navigation bar with tabs: "题目" (selected), "排名", "状态", and "提问". The main content area displays a message "CS101 / 题库 (包括计概、数算题目)". Below this, a section titled "#50229312 提交状态" shows the status as "Accepted". There are four buttons: "查看", "提交", "统计", and "提问".

#50229312 提交状态

状态: Accepted

源代码

```

n = int(input())
print(n)
for i in range(n):
 a,b,c=list(map(str,input().split()))
 Haab={"pop":1,"no":2,"zip":3,"zotz":4,"tzec":5,"xul":6,"yoxkin":7,"mol":8,"
 Tzolkin={1:"imix",2:"ik",3:"akbal",4:"kan",5:"chicchan",6:"cimi",7:"manik",
 _a = int(a.strip('.'))
 _c = int(c)
 sgm = 20*(Haab[b] - 1) + _a + _c*365
 y = sgm // 260
 m = Tzolkin[(sgm+1) % 20]
 d = sgm % 13 + 1
 print(d,m,y)

```

基本信息

#: 50229312  
题目: 01008  
提交人: 郭旭杰  
内存: 3680kB  
时间: 27ms  
语言: Python3  
提交时间: 2025-10-05 10:28:33

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

### 230B. T-primes (选做)

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

思路: 做了两个星期才通过

简而言之就是找质数的完全平方数。

总是 TLE, 后来改进了算法, 又用了 PyPy3, 还将第 63 次输入的数据单独讨论以节省时间才通过。

| My Submissions            |                                    |                |                              |           |                                                                 |         |          |
|---------------------------|------------------------------------|----------------|------------------------------|-----------|-----------------------------------------------------------------|---------|----------|
| #                         | When                               | Who            | Problem                      | Lang      | Verdict                                                         | Time    | Memory   |
| <a href="#">341555915</a> | Oct/03/2025 03:28 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 3-64 | <span style="color: green;">Accepted</span>                     | 1560 ms | 15200 KB |
| <a href="#">341555745</a> | Oct/03/2025 03:26 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 3-64 | <span style="color: red;">Wrong answer on test 17</span>        | 280 ms  | 15900 KB |
| <a href="#">341555307</a> | Oct/03/2025 03:21 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 3-64 | <span style="color: red;">Time limit exceeded on test 63</span> | 2000 ms | 15400 KB |
| <a href="#">341554733</a> | Oct/03/2025 03:16 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 3-64 | <span style="color: red;">Time limit exceeded on test 36</span> | 2000 ms | 14400 KB |
| <a href="#">341554520</a> | Oct/03/2025 03:14 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 3-64 | <span style="color: red;">Time limit exceeded on test 63</span> | 2000 ms | 15400 KB |
| <a href="#">341553163</a> | Oct/03/2025 03:00 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 3-64 | <span style="color: red;">Time limit exceeded on test 63</span> | 2000 ms | 14900 KB |
| <a href="#">341475865</a> | Oct/02/2025 16:54 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | PyPy 2    | <span style="color: red;">Runtime error on test 1</span>        | 154 ms  | 0 KB     |
| <a href="#">341475757</a> | Oct/02/2025 16:54 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 36</span> | 2000 ms | 13800 KB |
| <a href="#">340689082</a> | Sep/27/2025 12:06 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 36</span> | 2000 ms | 13800 KB |
| <a href="#">340689041</a> | Sep/27/2025 12:06 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Wrong answer on test 2</span>         | 92 ms   | 0 KB     |
| <a href="#">340688958</a> | Sep/27/2025 12:04 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Compilation error</span>              | 0 ms    | 0 KB     |
| <a href="#">340688745</a> | Sep/27/2025 12:01 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 33</span> | 2000 ms | 13800 KB |
| <a href="#">340688602</a> | Sep/27/2025 11:58 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 33</span> | 2000 ms | 13800 KB |
| <a href="#">340688418</a> | Sep/27/2025 11:55 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 36</span> | 2000 ms | 13900 KB |
| <a href="#">340688220</a> | Sep/27/2025 11:52 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 36</span> | 2000 ms | 13300 KB |
| <a href="#">340688071</a> | Sep/27/2025 11:49 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 36</span> | 2000 ms | 13400 KB |
| <a href="#">340688027</a> | Sep/27/2025 11:48 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Wrong answer on test 1</span>         | 92 ms   | 0 KB     |
| <a href="#">340687903</a> | Sep/27/2025 11:46 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 33</span> | 2000 ms | 13400 KB |
| <a href="#">340687814</a> | Sep/27/2025 11:44 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Wrong answer on test 2</span>         | 124 ms  | 100 KB   |
| <a href="#">340687468</a> | Sep/27/2025 11:38 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Wrong answer on test 4</span>         | 92 ms   | 0 KB     |
| <a href="#">340687398</a> | Sep/27/2025 11:37 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Wrong answer on test 4</span>         | 124 ms  | 0 KB     |
| <a href="#">340687314</a> | Sep/27/2025 11:35 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Wrong answer on test 4</span>         | 124 ms  | 0 KB     |
| <a href="#">340619601</a> | Sep/26/2025 21:13 <sup>UTC+8</sup> | LittleBeetroot | <a href="#">B - T-primes</a> | Python 3  | <span style="color: red;">Time limit exceeded on test 1</span>  | 2000 ms | 0 KB     |

代码

```
```python
```

```
from math import sqrt

def isprime(k):                      # 1个用法
    if k == 2 or k == 3:
        return True
    elif k == 1 or k == 4:
        return False
    else:
        blacklist_1 = 0
        for i in range(2, int(sqrt(k))+1):
            if k%i == 0:
                blacklist_1 = 1
                break
        if blacklist_1 == 1:
```

```
        return False
    else:
        return True

lines = int(input())
ns = list(map(int, input().split()))

for n in ns:

    if n == 999966000289:
        print("YES")
    elif n >= 5 and sqrt(n) - int(sqrt(n)) == 0 and n // 2 != 0:
        sq = int(sqrt(n))
        if isprime(sq):
            print("YES")
        else:
            print("NO")
    elif n != 4:
        print("NO")
    else:
        print("YES")

````
```

代码运行截图 <mark> (至少包含有"Accepted") </mark>

The screenshot shows a Codeforces contest submission page. At the top, there's a navigation bar with links like HOME, TOP, CATALOG, CONTESTS, GYM, PROBLEMSET, GROUPS, RATING, EDU, API, CALENDAR, HELP, and RAVAN. Below the navigation bar is a table showing the submission details:

| General   | #                           | Author    | Problem      | Lang     | Verdict    | Time     | Memory                 | Sent                   | Judged | Compare |
|-----------|-----------------------------|-----------|--------------|----------|------------|----------|------------------------|------------------------|--------|---------|
| 341555915 | Practice:<br>LittleBeetroot | 230B - 28 | PyPy<br>3-64 | Accepted | 1560<br>ms | 15232 KB | 2025-10-02<br>22:28:06 | 2025-10-02<br>22:28:06 |        |         |

Below the table is a code editor window titled "Source" containing Python code. The code checks if a number is prime and prints "YES" or "NO". It uses a blacklist of known non-prime numbers. The code is as follows:

```
from math import sqrt

def isprime(k):
 if k == 2 or k == 3:
 return True
 elif k == 1 or k == 4:
 return False
 else:
 blacklist_1 = 0
 for i in range(2, int(sqrt(k))+1):
 if k % i == 0:
 blacklist_1 = 1
 break
 if blacklist_1 == 1:
 return False
 else:
 return True

lines = int(input())
ns = list(map(int, input().split()))

for n in ns:
 if n == 999966000289:
 print("YES")
 elif n > 5 and sqrt(n) - int(sqrt(n)) == 0 and n // 2 != 0:
 sq = int(sqrt(n))
 if isprime(sq):
 print("YES")
 else:
 print("NO")
 elif n % 4:
 print("NO")
 else:
 print("YES")
```

At the bottom left of the code editor, there's a link "Click to see test details".

## ## 2. 学习总结和收获

通过本次学习，我掌握了基础的 **greedy** 算法，后三题还促使我开

始主动尝试优化代码。

我对 Hash Table 有了更深的认识，逐步灵活使用 Hash Table 映

射解题。

<mark>如果作业题目简单，有否额外练习题目，比如：OJ“计概 2025fall”每日选做、CF、LeetCode、洛谷

第一次参加 CodeForces 上的比赛，9 道题就做出来一题，取得+403 分，10847 名的成绩。

## Codeforces Hot News!

Wow! Coder LittleBeetroot competed in Squarepoint Challenge (Codeforces Round 1055, Div. 1 + Div. 2) and gained +403 rating points taking place 10847

Share it!

|       |                 |  |            |       |    |  |  |  |  |
|-------|-----------------|--|------------|-------|----|--|--|--|--|
| 11151 | cnyz_a          |  | <b>463</b> | 00:28 |    |  |  |  |  |
| 11151 | Vasu8725        |  | <b>463</b> | 00:28 | -1 |  |  |  |  |
| 11151 | hemalekharames  |  | <b>463</b> | 00:28 |    |  |  |  |  |
| 11151 | LittleBeetroot  |  | <b>463</b> | 00:28 |    |  |  |  |  |
| 11151 | ljgs68          |  | <b>463</b> | 00:28 |    |  |  |  |  |
| 11151 | rishit.aggarwal |  | <b>463</b> | 00:28 |    |  |  |  |  |
| 11151 | grishma5472     |  | <b>463</b> | 00:28 |    |  |  |  |  |

CODEFORCES Sponsored by TON

LittleBeetroot | Logout

HOME TOP CATALOG CONTESTS GYM PROBLEMSET GROUPS RATING EDU API CALENDAR HELP RAYAN

PROBLEMS SUBMIT CODE MY SUBMISSIONS STATUS HACKS ROOM STANDINGS CUSTOM INVOCATION

General

| #         | Author                     | Problem    | Lang      | Verdict  | Time   | Memory  | Sent                | Judged              |         |
|-----------|----------------------------|------------|-----------|----------|--------|---------|---------------------|---------------------|---------|
| 341678358 | Contestant: LittleBeetroot | 2152A - 32 | PyPy 3-64 | Accepted | 156 ms | 3268 KB | 2025-10-03 18:03:50 | 2025-10-03 20:43:41 | Compare |

→ Source

```
lines = int(input())
for i in range(lines):
 n = int(input())
 js = list(map(int, input().split()))
 jt = []
 for j in js:
 if j not in jt:
 jt.append(j)
 p = len(jt)
 print(2**p-1)
```

Click to see test details

本人练习了 sunnywhy 上的基础题，做出了冒泡排序这个经典问题。

2026考研算法：全程训练营（初试&机试）已经上线：<https://sunnywhy.com/camp/3415>，适合包括「浙大、复旦、上交、华师、中科大计算机&软件」等上机难度院校，也适合「难度友好型」院校。

题目描述

通过数 4550 提交数 12652 准度 显示标签

注意：提交的代码字符串中，不能出现 `sort`，请自己实现自己的排序函数，不要调用语言标准库的排序函数。

输入描述

第一行一个整数n（ $1 \leq n \leq 100$ ），表示需要输入的正整数的个数；  
第二行为用空格隔开的n个正整数（每个正整数均不超过100）。

输出描述

输出一行，表示输入的n个正整数，整数间用一个空格隔开，行末不允许有多余的空格。

样例1

输入 5  
2 8 5 1 3

输出 15

代码示例

```
1 n=int(input())
2 [x:=list(map(int, input().split()))]
3
4 def up(seet):
5 blacklist_1 = 0
6 if len(seet) == 1:
7 return True
8 else:
9 for _ in range(0, len(seet)-1):
10 if seet[_] > seet[_+1]:
11 blacklist_1 += 1
12 if blacklist_1 == 0:
13 return True
14 else:
15 return False
16
17 while not up():
18 for i in range(0, n-1):
19 if js[i] > js[i+1]:
20 sett = js[i]
21 js[i] = js[i+1]
22 js[i+1] = sett
23
24 print(*js, sep=' ')
```

测试输入

提交时间 结果 时长(ms) 语言

|                     |      |   |        |
|---------------------|------|---|--------|
| 2025-10-05 15:02:33 | 完美通过 | 0 | Python |
| 2025-10-05 01:21:47 | 完美通过 | 0 | Python |