

assignment4

Assignment #4: T-primes + 贪心

Updated 18:14 GMT+8 Sep 30, 2025

2025 fall, Complied by 顾桂榕-基础医学院



顾桂榕 医学预科办

说明:

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://typoraio.cn> 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。

3. 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：TV sets: $n(i \rightarrow a_i)$

carry most: $\leq m$

这题一开始没想清楚，其实Bob可以不全搬

代码

```
n, m = map(int, input().split())
price_list = list(map(int, input().split()))
neg_list = []
for i in price_list:
    if i <= 0:
        neg_list.append(i)
sorted_neg = sorted(neg_list)
if len(sorted_neg) <= m:
    gain = -sum(sorted_neg)
else:
    gain = -sum(sorted_neg[0:n])
print(gain)
```

代码运行截图 (至少包含有"Accepted")

By Rose-Rong, contest: Codeforces Beta Round 34 (Div. 2), problem: (B) Sale, Accepted, #, Copy

```
n, m = map(int, input().split())
price_list = list(map(int, input().split()))
neg_list = []
for i in price_list:
    if i<=0:
        neg_list.append(i)
sorted_neg = sorted(neg_list)
if len(sorted_neg)<= m:
    gain = -sum(sorted_neg)
else:
    gain = -sum(sorted_neg[0:m])
print(gain)
```

160A. Twins

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

思路：这题按照题目意思做就行了

代码

```
n = int(input())
value_list = list(map(int, input().split()))
sorted_list = sorted(value_list, reverse=True)
total = sum(value_list)
me = 0
i = 0
while True:
    if me > (total - me):
        break
    else:
        me += sorted_list[i]
        i += 1
print(i)
```

代码运行截图 (至少包含有"Accepted")

By Rose-Rong, contest: Codeforces Round 111 (Div. 2), problem: (A) Twins, Accepted, #, [Copy](#)

```
n = int(input())
value_list = list(map(int, input().split()))
sorted_list = sorted(value_list, reverse=True)
total = sum(value_list)
me = 0
i = 0
while True:
    if me > (total - me):
        break
    else:
        me += sorted_list[i]
        i += 1
print(i)
```

1879B. Chips on the Board

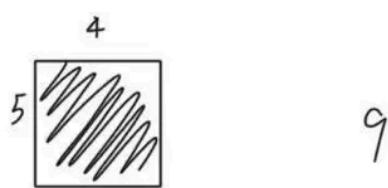
constructive algorithms, greedy, 900,

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

思路：

	1	4	1
3			
2			
2			

10



9

	4	5
2		
3	(2,1)	(2,2)

13

	5	2	43	52	31
3					
43					
32					
21					
25					

对于greedy题目来说，思路还是很重要的

代码

```
test_cases = int(input())
for _ in range(test_cases):
    n = int(input())
    a_list = list(map(int, input().split()))
    b_list = list(map(int, input().split()))
    a_min = min(a_list)
    b_min = min(b_list)
    sum_a = a_min*n + sum(b_list)
    sum_b = b_min*n + sum(a_list)
    print(min(sum_a, sum_b))
```

代码运行截图 (至少包含有"Accepted")

By Rose-Rong, contest: Educational Codeforces Round 155 (Rated for Div. 2), problem: (B) Chips on the Board, Accepted, #, [Copy](#)

```
test_cases = int(input())
for _ in range(test_cases):
    n = int(input())
    a_list = list(map(int, input().split()))
    b_list = list(map(int, input().split()))
    a_min = min(a_list)
    b_min = min(b_list)
    sum_a = a_min*n + sum(b_list)
    sum_b = b_min*n + sum(a_list)
    print(min(sum_a, sum_b))
```

Mo1o17: 装箱问题

greedy, <http://cs1o1.openjudge.cn/pctbook/Mo1o17/> 

思路：逻辑爆炸的一道题，WA了很多次。

很考验思维的完善性和严谨性。

代码

```
import math
while True:
    s = list(map(int, input().split()))
    ans = 0
    if s==[0,0,0,0,0,0]:
```

```

        break
ans+=s[5]+s[4]+s[3]
if s[4]>0:
    if s[0] ≤ 11*s[4]:
        s[0]=0
    else:
        s[0]-=11*s[4]
if s[3]>0:
    if s[1] ≤ 5*s[3]:
        s[1]=0
        s[0]=max(s[0]-(s[3]*20-s[1]*4),0)
    else:
        s[1]-=5*s[3]
if s[2]>0:
    ans+=math.ceil(s[2]/4)
    if s[2]%4==1:
        if s[1]==0:
            s[0]=max(0,s[0]-27)
        if s[1] ≤ 5 and s≠0:
            s[1]=0
            if s[0]>7:
                ans += math.ceil((s[0]-7)/36)
            s[0]=0
        else:
            s[1]-=5
            s[0]=max(0,s[0]-7)
    elif s[2]%4==2:
        if s[1]==0:
            s[0]=max(0,s[0]-18)
        if s[1] ≤ 3 and s≠0:
            s[1] = 0
            if s[0]>6:
                ans += math.ceil((s[0]-6)/36)
            s[0]=0
        else:
            s[1]-=3
            s[0]=max(0,s[0]-6)
    elif s[2]%4==3:
        if s[1]==0:
            s[0]=max(0,s[0]-9)

```

```

        if s[1] ≤ 1 and s ≠ 0:
            s[1] = 0
            if s[0] > 5:
                ans += math.ceil((s[0]-5)/36)
            s[0]=0
        else:
            s[1] -= 1
            s[0]=max(0,s[0]-5)

    if s[1]>0:
        ans += math.ceil(s[1] / 9)
        remain = 36-(s[1]-math.floor(s[1]/9)*9)*4
        if s[0]>remain:
            ans+=math.ceil((s[0]-remain)/36)
print(ans)

```

代码运行截图 (至少包含有"Accepted")

#50259575提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```

import math
while True:
    s = list(map(int, input().split()))
    ans = 0
    if s==[0,0,0,0,0,0]:
        break
    ans+=s[5]+s[4]+s[3]
    if s[4]>0:
        if s[0]≤11*s[4]:
            s[0]=0
        else:
            s[0]-=11*s[4]
    if s[3]>0:
        if s[1]≤5*s[3]:
            s[0]=max(s[0]-(s[3]*20-s[1]*4),0)
            s[1]=0
        else:
            s[1]-=5*s[3]
    if s[2]>0:
        ans+=math.ceil(s[2]/4)
        if s[2]%-4==1:
            if s[1]==0:
                s[0]=max(0,s[0]-27)
            elif 0<s[1]≤5:
                s[1]=0
            if s[0]>7:
                ans += math.ceil((s[0]-7)/36)
                s[0]=0
        else:

```

基本信息

#: 50259575
 题目: M01017
 提交人: Rose-Rong
 内存: 3792kB
 时间: 36ms
 语言: Python3
 提交时间: 2025-10-08 15:11:59

M01008: Maya Calendar

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

思路：

Calendar.1.o(Haab-19months):

[18 * 20]pop, no, zip, zotz, tzec, xul, yoxkin, mol, chen, yax, zac, ceh, mac, kankin, muan, pax, koyab, cumhu(0-19)

[5]0, 1, 2, 3, 4

Calendar.2.o(Tzolkin-thirteen periods * 20days):

20 names: imix, ik, akbal, kan, chicchan, cimi, manik, lamat, muluk, ok, chuen, eb, ben, ix, mem, cib, caban, eznab, canac, ahau

13 numbers(1-13)

代码

```
import math
test_cases = int(input())
Haab_Calendar = {'pop':0, 'no':1, 'zip':2, 'zotz':3, 'tzec':4,
'xul':5, 'yoxkin':6, 'mol':7, 'chen':8, 'yax':9, 'zac':10,
'ceh':11, 'mac':12, 'kankin':13, 'muan':14, 'pax':15, 'koyab':16,
'cumhu':17, 'uayet':18}
Tzolkin_Calendar_names =['imix', 'ik', 'akbal', 'kan',
'chicchan', 'cimi', 'manik', 'lamat', 'muluk', 'ok', 'chuen',
'eb', 'ben', 'ix', 'mem', 'cib', 'caban', 'eznab', 'canac',
'ahau']
print(test_cases)
for i in range(test_cases):
    NumberOfTheDay, Month, Year = list(input().split())
    NumberOfTheDay=NumberOfTheDay.strip('.')
    days =
    int(Year)*365+Haab_Calendar[Month]*20+int(NumberOfTheDay)
    year = math.floor(days/260)
    number = days%13+1
    if number==0:
        number=13
    name = Tzolkin_Calendar_names[days%20]
    print(f'{number} {name} {year}')
```

代码运行截图 (至少包含有"Accepted")

#50260530提交状态

查看

提交

统计

状态: Accepted

源代码

```
import math
test_cases = int(input())
Haab_Calendar = {'pop':0, 'no':1, 'zip':2, 'zotz':3, 'tzec':4, 'xul':5, 'yoxk':6, 'ch'en':7, 'yax':8, 'zac':9, 'ceh':10, 'mac':11, 'kankin':12, 'pax':13, 'koyab':14, 'cumhu':15, 'uayet':16}
Tzolkin_Calendar_names =['imix', 'ik', 'akbal', 'kan', 'chicchan', 'cimi', 'manik', 'ok', 'chuen', 'eb', 'ben', 'ix', 'mem', 'cib', 'caban']
print(test_cases)
for i in range(test_cases):
    NumberOfTheDay,Month,Year = list(input().split())
    NumberOfTheDay=NumberOfTheDay.strip('.')
    days = int(Year)*365+Haab_Calendar[Month]*20+int(NumberOfTheDay)
    year = math.floor(days/260)
    number = days%13+1
    if number==0:
        number=13
    name = Tzolkin_Calendar_names[days%20]
    print(f'{number} {name} {year}')
```

基本信息

#: 50260530

题目: 01008

提交人: Rose-Rong

内存: 3704kB

时间: 25ms

语言: Python3

提交时间: 2025-10-08 15:51:20

230B. T-primes (选做)

binary search, implementation, math, number theory, 1300,

<http://codeforces.com/problemset/problem/230/B>

思路:

10月8日下午有事，原代码已编写成功，但超时。有时间学习一下筛法。

代码

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2025fall每日选做”、CF、
LeetCode、洛谷等网站题目。

9月25日笔记：集合与列表

**1. 集合 (set)

(1)创建集合：

```
my_set = {1,2,3,4,5}
#如果写my_set = {}是不对的，这会使计算机认为是字典
#创建空集合应该这样
empty_set = set()
```

(2)添加元素：

```
n = 'mnpq'
my_set.add(6)
my_set.update([7,8,9])→ {1, 2, 3, 4, 5, 6, 7, 8, 9}
my_set.update(7,8,9)→x
my_set.update('m','n')=my_set.update(['m','n'])→{1,2,3,4,5,6,'m','n'}
my_set.update(m,n)→没定义的话是x的
my_set=set(n)→{'m', 'q', 'p', 'n'}排列组合（随机）
```

(3)删除元素：

```
my_set.remove(3)
my_set.discard(10)
# remove和discard的区别：
# 若元素不存在remove会报错，discard不会
popped_element = my_set.pop()
# 随机删除一个元素并取出
```

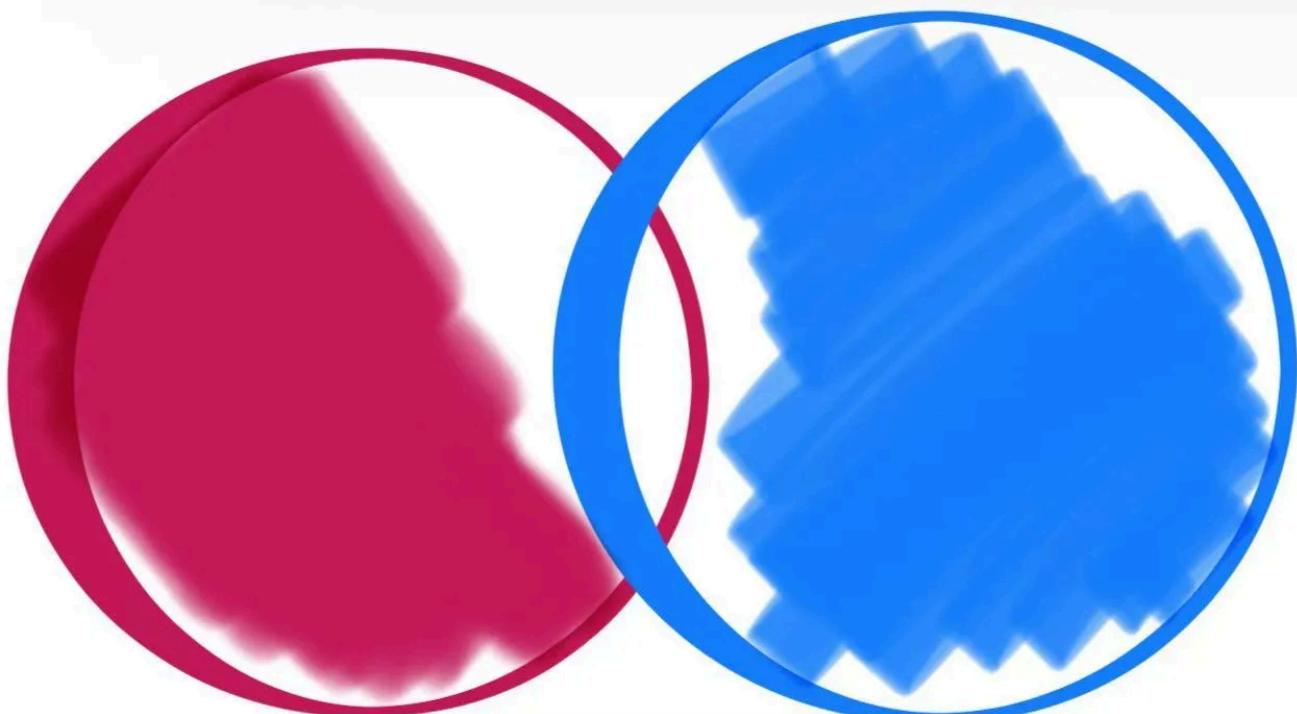
(4)检查元素是否存在：

```
if 5 in my_set:
len(my_set)
```

(5)集合运算：

并集: union_set = set1 | set2 或 set1.union(set2)
交集: intersection_set = set1 & set2 或 set1.intersection(set2)
差集: difference_set = set1 - set2 或 set1.difference(set2)

```
# ① A={1,2,5} B={1,2,3,4}
    difference_set = {1,2,5}
# ② A={1,2} B={3,4}
    difference_set = {1,2}
对称差集: symmetric_difference = set1 ^ set2 或
set1.symmetric_difference(set2)
```



(6)集合推导式:

没有100

```
squares = {x**2 for x in range(10)}
→{0, 1, 4, 9, 16, 25, 36, 49, 64, 81} 可能错排
```

**2. 集合(*Dictionary*) (字典没有统计长度)

(1)添加/修改元素:

```
my_dict['occupation']='manager'
my_dict.update({'country':'China','gender':'female'})
# 多个或一个键值对
my_dict['age']=26(modify)
```

(2)访问元素:

```
print(my_dict['occupation']) → manager
print(my_dict.get('country', 'unknown'))
# former key不存在会报错
# latter key不存在返回默认值
→ unknown(预备括号里的值)
# 没有预备括号返回None
```

(3)删除元素：

```
del my_dict['city']
# delete key & value together
value = my_dict.pop('occupation')
→ 返回manager
key, value = my_dict.popitem()
# 删除并返回最后一个键值对
```

(4)获取所有的键值对：

```
keys = my_dict.keys()
values = my_dict.values()
items = my_dict.items()
# print(keys)
→ dict_keys(['occupation'])
# print(*keys)
→ occupation
# print(list(my_dict.keys()))
→ ['q', 't']
```

(5)其他：

```
# 清空字典：
my_dict.clear()
```

(6)字典推导式：

```
squares_dict = {x:x**2 for x in range(1,6)}
→ {1:1, 2:4, 3:9, 4:16, 5:25}
```

```
# 过滤字典
original_dict = {'a':1,'b':2,'c':3,'d':4}
filtered_dict = {k:v for k,v in original_dict.items() if v>2}
→{'c':3,'d':4}
```

**3.应用

```
# 统计单词频率
text = "apple banana apple orange banana apple"
words = text.split()→['apple', 'banana', 'apple', 'orange',
'banana', 'apple']
word_count = {}

for word in words:
    word_count[word] = word_count.get(word, 0) + 1

print(word_count) # {'apple': 3, 'banana': 2, 'orange': 1}

# 使用 collections.Counter 更简单
from collections import Counter
word_count = Counter(words)
print(word_count) # Counter({'apple': 3, 'banana': 2, 'orange':
1})
```

word_count.get(word, 0)→这是在取对应单词的已出现次数

对于一个字符串如：

```
alphabet = list('ABCD')
→['A','B','C','D']
list("apple banana apple orange banana apple")
→['a', 'p', 'p', 'l', 'e', ' ', 'b', 'a', 'n', 'a', 'n', 'a', ' ',
```

```
# 成绩转换
grade_conversion = {'A': 4.0, 'B': 3.0, 'C': 2.0, 'D': 1.0, 'F':
0.0}
grades = ['A', 'B', 'C', 'A', 'B']
```

```
gpa = sum(grade_conversion[grade] for grade in grades) / len(grades)
print(f"平均绩点: {gpa}") # 平均绩点: 3.0
```



9月27日笔记

02724生日相同

#50157184提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
n = int(input())
birthday_dict = {}
birthday_date = []
for i in range(n):
    id_num,month,day = input().split()
    month = int(month)
    day = int(day)
    if (month,day) not in birthday_date:
        birthday_date.append((month,day))
        birthday_dict[month,day] = []
    birthday_dict[month,day].append(id_num)
sorted_date = sorted(birthday_dict.keys())
for i in sorted_date:
    j = birthday_dict[i]
    if len(j)>1:
        s = str(i[0]) + ' ' +str(i[1])+' '+' '.join(j)
        print (s)
```

基本信息

#: 50157184
题目: E02724
提交人: Rose-Rong
内存: 3616kB
时间: 21ms
语言: Python3
提交时间: 2025-09-27 11:25:49

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

```
n = int(input())
birthday_dict = {}
birthday_date = []
for i in range(n):
    id_num,month,day = input().split()
    month = int(month)
    day = int(day)
    if (month,day) not in birthday_date:
        birthday_date.append((month,day))
        birthday_dict[month,day] = []
    birthday_dict[month,day].append(id_num)
sorted_date = sorted(birthday_dict.keys())
for i in sorted_date:
```

```

j = birthday_dict[i]
if len(j)>1:
    s = str(i[0]) +' '+str(i[1])+ ' '+ .join(j)
    print (s)

```

***注意：

(1) 对元组进行排序（升序），需要将数字的类型由字符串转化为整型
eg: (10, 1) & (2,1)若为"str"型则会一个数字接一个数字比较

(2) 最后输出时需要将元组的括号拆掉

Method(1):

```

i = (2,11)
a,b = i
print(str('a'+' '+str('b'))
→2 11

```

Method(2):

```
print(str(i[0])+' '+str(i[1]))
```



06374文字排版

#50157794提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```

n = int(input())
l = input().split()
ans = []
tmp = l[0] + ' '
for i in l[1:]:
    if len(tmp)+len(i)>80:
        ans.append(tmp.rstrip())
        tmp = i + ' '
    else:
        tmp += i + ' '
else:
    ans.append(tmp.rstrip())
print('\n'.join(ans))

```

基本信息

#: 50157794
题目: E06374
提交人: Rose-Rong
内存: 3600kB
时间: 20ms
语言: Python3
提交时间: 2025-09-27 12:26:40

1. 常用的命名:

- idx → index (索引)
- cnt → count (计数)
- len → length (长度)
- str → string (字符串)
- arr → array (数组)
- buf → buffer (缓冲区)
- ptr → pointer (指针)
- tmp → temporary (暂时存储)
- ans → answer (最终结果)
- res → result

2.map的用法:

map接收一个函数和一个可迭代对象（如列表）作为参数，然后将函数应用于可迭代对象中的每一个元素

(1)最常用的：

a,b,c = map(int,input().split())

将a,b,c都转换为接收结果的整型

对于这道题来说没必要：

只要用input().split()即可

l = input().split()

input sample: I love ice-cream.

→ ['I', 'love', 'ice-cream.']}

这个地方ice-cream后面的'.'也在元素中

input sample:I love ice-cream

['I','love','ice-cream']

(2)处理多个迭代对象：

list(map(lambda x, y : x+y,[1,2,3],[4,5,6]))

→[5,7,9]

list(map(lambda x:x**2,[1,2,3]))

→[1,4,9]

3.strip的用法：（仅适用于字符串）

.rstrip()-移除末尾指定字符

.lstrip()-移除开头指定字符

.strip()-同时移除开头结尾指定字符

tmp在添加列表中元素的时候没有携带"号

- 如果不提供参数， 默认移除末尾的空白字符（空格、制表符、换行符等）

```
text = "hello world    \t\n"
print(text.rstrip()) # 输出: "hello world"

text2 = "hello world    "
print(text2.rstrip()) # 输出: "hello world"
```

- 如果提供 chars 参数（用'')，则移除末尾指定的字符（这时这个指定字符必须在末尾）

eg:

```
tmp = 'I love ice-cream.'
print(tmp.rstrip('ice-cream'))
→I love ice-cream. (无效)

tmp = 'I love ice-cream.'
print(tmp.rstrip('ice-cream.')) 
→I love
tmp = 'I love ice-cream.'
print(tmp.rstrip('.'))
→I love ice-cream
```



02689大小写字母互换

```

s = input()
gap = ord('a') - ord('A')

ans = []
for i in s:
    if 'A' <= i <= 'Z':
        ans += chr(ord(i) + gap)
    elif 'a' <= i <= 'z':
        ans += chr(ord(i) - gap)
    else:
        ans += i

print(''.join(ans))

```

**chr() & ord()

两者都为内置函数，且互补

- ord(character): 字符 → Unicode 码点 (整数)
- chr(integer): Unicode 码点 (整数) → 字符
- 大写字母>小写字母

#50150006提交状态

[查看](#)

[提交](#)

[统计](#)

[提问](#)

状态: Accepted

源代码

```

n = list(input())
u = list('ABCDEFGHIJKLMNOPQRSTUVWXYZ')
l = list('abcdefghijklmnopqrstuvwxyz')
for i in range(len(n)):
    if n[i] in u:
        n[i]=l[u.index(n[i])]
    elif n[i] in l:
        n[i]=u[l.index(n[i])]
print(*n,sep='')

```

基本信息

#: 50150006
 题目: E02689
 提交人: Rose-Rong
 内存: 3612kB
 时间: 19ms
 语言: Python3
 提交时间: 2025-09-26 19:21:57



12556: 编码字符串

状态: Accepted

源代码

```
a = list(input().lower())
s = []
b = 1
for i in range(len(a)):
    if i == len(a)-1:
        break
    if a[i] == a[i+1]:
        b += 1
    else:
        s.append((a[i],b))
        b = 1
s.append((a[len(a)-1],b))
ans = ''
for i,j in s:
    ans += f'({i},{j})'
print(ans)
```

基本信息

#: 50159827
 题目: E12556
 提交人: Rose-Rong
 内存: 3644kB
 时间: 19ms
 语言: Python3
 提交时间: 2025-09-27 14:52:43

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

**主要是output的形式要注意

*s,sep=""

→可以去除列表外的[],但是直接打印元组，所有字母带''

ans = ''

for i,j ins:

ans += f'({i},{j})'

→sample answer



996A. Hit the Lottery

By Rose-Rong, contest: Codeforces Round 492 (Div. 2) [Thanks, uDebug!], problem: (A) Hit the Lottery, Accepted, #, Copy

```
n = int(input())
num = [100, 20, 10, 5, 1]
ans = 0
for i in range(len(num)):
    if n%num[i]==0:
        ans += n//num[i]
        break
    while num[i]<=n:
        a = n//num[i]
        ans += a
        n -= a*num[i]
    i += 1
print(ans)
```



9月28日笔记

19949: 提取实体

#50169034提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
n = int(input())
ans = 0
for i in range(n):
    s = list(input().split())
    for j in range(len(s)):
        if j == len(s)-1:
            break
        if '#' in s[j] and '#' not in s[j+1]:
            ans += 1
    if '#' in s[len(s)-1]:
        ans += 1
print(ans)
```

基本信息

#: 50169034
题目: E19949
提交人: Rose-Rong
内存: 3604kB
时间: 19ms
语言: Python3
提交时间: 2025-09-28 09:25:31

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于



01218: THE DRUNK JAILER

Version 1

思路类似“校园外的树”，建立一个由单元素(o)组成的列表，根据题目要求迭代

#50169829提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
test_cases = int(input())
for _ in range(test_cases):
    n = int(input())
    cells = [0] * n
    for round_num in range(1, n+1):
        for i in range(round_num - 1, n, round_num):
            cells[i] = 1 - cells[i]
    escaped_cells = sum(cells)
    print(escaped_cells)
```

基本信息

#: 50169829
题目: E01218
提交人: Rose-Rong
内存: 3520kB
时间: 24ms
语言: Python3
提交时间: 2025-09-28 10:50:38

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

```
test_cases = int(input())
for _ in range(test_cases):
    n = int(input())
    cells = [0] * n
    for round_num in range(1, n+1):
        for i in range(round_num - 1, n, round_num):
            cells[i] = 1 - cells[i]
```

```
escaped_cells = sum(cells)
print(escaped_cells)
```

cells[i] = 1 - cells[i] 这一步很巧妙，直接完成了[0]&[1]的转换

for round_num in range(1,n+1):

round_num指轮数,这是在完成n轮迭代

for i in range(round_num - 1,n,round_num):

round_num - 1:start index

n-1:end index

round_num:step

Version 2(开关灯及牢房逃脱的通用数学解法)

#50169865提交状态

查看

提交

统计

提问

状态: Accepted

源代码

```
import math
test_cases = int(input())
for _ in range(test_cases):
    n = int(input())
    escaped_cells = math.floor(n**(.5))
    print(escaped_cells)
```

基本信息

#: 50169865

题目: E01218

提交人: Rose-Rong

内存: 3564kB

时间: 19ms

语言: Python3

提交时间: 2025-09-28 10:54:27

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

```
import math
test_cases = int(input())
for _ in range(test_cases):
    n = int(input())
    escaped_cells = math.floor(n**(.5))
    print(escaped_cells)
```

对于任意一个数 ($n \neq 1$)，首先1&n是它的因数

若还有其他因数，则必定成对出现：

eg:

$6 = 2 \times 3 (1, 2, 3, 6)$

也就是说这个数会被迭代4次($n = 1 \rightarrow 2 \rightarrow 3 \rightarrow 6$)

那么任意一个数若不为平方数，就会被迭代偶数次，回到最初状态[0]→lock

所以只有平方数的prisoner能escape

列举规律如下：

1	$n=1$	
2	$n=2$	1 1 1 0
3	$n=3$	1 1 1 1 0 1 1 0 0
2	$n=4$	1 1 1 1 1 0 1 0 1 0 0 0 1 0 0 1
2	$n=5$	1 1 1 1 1 1 0 1 0 1 1 0 0 1 1 1 0 0 0 1 1 0 0 0 0
2	$n=6$	1 1 1 1 1 1 1 0 1 0 1 0 1 0 0 1 1 1 1 0 0 0 1 1 1 0 0 0 0 1 1 0 0 0 0 0
2	$n=7$	1 1 1 1 1 1 1 1 0 1 0 1 0 1 1 0 0 1 1 1 1 1 0 0 0 1 1 1 1 0 0 0 0 1 1 1 0 0 0 0 0 1 1 0 0 0 0 0 0
3	$n=9$	1 1 1 1 1 1 1 1 1 1 0 1 0 1 0 1 0 1 1 0 0 1 1 1 1 1 0 1 0 0 0 1 1 1 1 0 1 0 0 0 0 1 1 1 0 1 0 0 0 0 0 1 1 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1



1352A. Sum of Round Numbers

General

#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged
340835180	Practice: Rose-Rong	1352A - 17	Python 3	Accepted	140 ms	36 KB	2025-09-28 07:08:56	2025-09-28 07:09:12

→ Source

```
test_cases = int(input())
ans = []
for _ in range(test_cases):
    num = int(input())
    num_list = list(str(num))
    leng = len(str(num))
    if num in range(0, 10) or num%(10**leng-1) == 0:
        print(1)
        print(num)
    else:
        a = len(str(num)) - num_list.count('0')
        print(a)
        for i in range(leng):
            if num_list[i] != '0':
                ans.append(int(num_list[i])*(10**(leng-i-1)))
print(*ans)
ans.clear()
```

最后列表添加元素傻了，应该用append



18223: 24点

Version 1

状态: Accepted

源代码

```
test_cases = int(input())
ans = []
for _ in range(test_cases):
    a,b,c,d = map(int,input().split())
    for i in [a,-a]:
        for j in [b,-b]:
            for k in [c,-c]:
                for l in [d,-d]:
                    tmp = i+j+k+l
                    ans.append(tmp)
print("YES" if 24 in ans else "NO")
ans.clear()
```

基本信息

#: 50173006
 题目: E18223
 提交人: Rose-Rong
 内存: 3600kB
 时间: 21ms
 语言: Python3
 提交时间: 2025-09-28 15:00:13

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

```
test_cases = int(input())
ans = []
for _ in range(test_cases):
    a,b,c,d = map(int,input().split())
    for i in [a,-a]:
        for j in [b,-b]:
            for k in [c,-c]:
                for l in [d,-d]:
                    tmp = i+j+k+l
                    ans.append(tmp)
print("YES" if 24 in ans else "NO")
ans.clear()
```

这里用到了 $[a, -a]$ 的迭代（从两个元素中选）

最后一步可以这么写：

```
if tmp == 24:
flag = True
print('YES' if flag else 'NO')
就不用再向列表里添加元素占内存了
```

Version 2

```
import itertools
test_cases = int(input())
for _ in range(test_cases):
```

```

a,b,c,d = map(int,input().split())
res = a+b+c+d
data = set(itertools.permutations([a,b,c,d],4))

for i in data:
    res1 = i[0] + i[1] + i[2] - i[3]
    res2 = i[0] - i[1] - i[2] - i[3]
    res3 = i[0] - i[1] + i[2] - i[3]
    if res == 24 or res1 == 24 or res2 == 24 or res3 == 24:
        print('YES')
        break
    else:
        print('NO')

```

(1)permutations(from itertools)排列

& combinations(from itertools)组合: (内存小, 高效)

import itertools
itertools.combinations([a,b,c],2)
itertools.permutations([a,b,c,d],4)
itertools.combinations_with_replacement([a,b,c],2)

→会出现 (a,a)

4表示4个数进行全排列

如果是3的话就是在4个元素中选3个进行排列

但可能有重复情况:

例如[1,1,3,4]输出的实际上是不重复情况的2倍

因此采用set去重

这里set{}内的元素是多个元组()

只去除重复元组, 不去除元组中的重复元素

另: 计算排列组合数-math.perm/comb(比统计len(上述函数)快很多)

```

import math
math.perm(4,3)→A43
math.comb(4,3)→C43

```

(2)符号的选择

使用permutations已经将数字的不同排列存储至data

考虑符号时仅考虑3个符号即可

(因为若 $i[0] \rightarrow -i[0]$ 其实已经在 $i[0]$ 成为第2 or 3 or 4元素时实现过了)

那便有4种：

全+， 全-， 1+2-， 2+1-

Version 3

看大佬用了random(类蒙特卡洛模拟)

```
import random

# 读取测试组数
num_tests = int(input())

for _ in range(num_tests):
    # 读取四个整数
    numbers = list(map(int, input().split()))

    # 构建每个数的正负选项 [[a, -a], [b, -b], [c, -c], [d, -d]]
    pos_neg_options = []
    for num in numbers:
        pos_neg_options.append([num, -num])

    # 存储随机模拟的结果
    simulation_results = []

    # 进行 256 次随机模拟
    for _ in range(256):
        total = 0
        # 对每个数，随机选择正或负
        for i in range(4):
            total += random.choice(pos_neg_options[i])
        simulation_results.append(total)

    # 判断 24 是否在模拟结果中出现
    print('YES' if 24 in simulation_results else 'NO')
```

random

```
import random
numbers = [0,1,2,3,4]
1.random.choice(numbers)
pos_neg_options = ([-1,1],[-2,2],[-3,3])
for i in range(4):
    total += random.choice(pos_neg_options[i])
```

2. 常用random函数

- (1).print(random.random())→生成[0.0,1.0]随机浮点数
- (2).print(random.randint(1,10))→生成[a,b]的随机整数
- (3).print(random.uniform(1.0,5.0))→生成[a,b]的随机浮点数

(4).打乱序列顺序

```
cards = ['A','B','C','R']
random.shuffle(cards)
print(cards)→random result
(5).从整体中无放回抽取 k 个样本
lottery = random.sample(range(1,50),6)
从[1,50](range特性)中抽k个
```

Version 4:

```
import operator
m = int(input())
op = ["++-", "--", "++", "+--", "-+-", "+--", "-+-", "--+"]
op_dict = {"+": operator.add, "-": operator.sub}

output = []
for i in range(m):
    a, b, c, d = [int(a) for a in input().split()]
    for o in op:
        if abs(op_dict[o[2]](op_dict[o[1]](op_dict[o[0]](a, b),
c), d)) == 24:
            output.append("YES")
            break
    else:
        output.append("NO")
```

```
print('\n'.join(output))
```

operator.add & operator.sub

```
import operator  
operator.add(a,b)→a+b  
operator.sub(a,b)→a-b
```

```
for o in op:  
    step1 = op_dict[op[0]](a,b)  
    step2 = op_dict[op[1]](step1,c)  
    step3 = op_dict[op[2]](step2,d)  
    if abs(step3)==24:  
        ans.append('YES')  
        break  
    else:  
        ans.append('NO')
```

abs的原因：

所给代码只能是a()b()c()d,不能在a前加'-',所以加上abs符号



9月30日笔记

1335A. Candies and Two Sisters

By Rose-Rong, contest: Codeforces Round 634 (Div. 3), problem: (A) Candies and Two Sisters, Accepted, #, [Copy](#)

```
t = int(input())  
for _ in range(t):  
    n = int(input())  
    if n%2 == 0:  
        ans = n//2 - 1  
    else:  
        ans = (n-1)//2  
    print(ans)
```

```
t = int(input())  
for _ in range(t):  
    n = int(input())  
    if n%2 == 0:
```

```
    ans = n//2 - 1
else:
    ans = (n-1)//2
print(ans)
```



03143: 验证“歌德巴赫猜想”

状态: Accepted

源代码

```
import math
l = list(range(2,2000))
for _ in range(2,2000):
    for j in range(2,_):
        if __%j==0:
            l.remove(_)
            break
x = int(input())
if x<6 or x%2 != 0:
    print('Error!')
else:
    for i in l:
        if i>math.floor(x/2):
            break
        if (x-i) in l and x-i>=i:
            print(f'{x}={i}+{x-i}')
```

基本信息

#: 50197192
题目: E03143
提交人: Rose-Rong
内存: 3636kB
时间: 51ms
语言: Python3
提交时间: 2025-09-30 16:21:54

```
import math
l = list(range(2,2000))
for _ in range(2,2000):
    for j in range(2,_):
        if __%j==0:
            l.remove(_)
            break
x = int(input())
if x<6 or x%2 ≠ 0:
    print('Error!')
else:
    for i in l:
        if i>math.floor(x/2):
            break
        if (x-i) in l and x-i≥i:
            print(f'{x}={i}+{x-i}')
```





151A. Soft Drinking

friends:n

soft drinks:k l ml

limes:c d slices

salt:p (g)

toast: n(person) * (nl ml soft drinks&lime(1 slice)&salt(np grams))

By Rose-Rong, contest: Codeforces Round 107 (Div. 2), problem: (A) Soft Drinking, Accepted, #, Copy

```
n, k, l, c, d, p, nl, np = map(int, input().split())
drinks = (k*l)//(n*nl)
limes = (c*d)//n
salt = p//(n*np)
s = min(drinks, limes, salt)
print(s)
```

```
n, k, l, c, d, p, nl, np = map(int, input().split())
drinks = (k*l)//(n*nl)
limes = (c*d)//n
salt = p//(n*np)
s = min(drinks, limes, salt)
print(s)
```



19944: 这一天星期几

状态: Accepted

源代码

```

import math
num_weekday = {1:'Monday', 2:'Tuesday', 3:'Wednesday', 4:'Thursday', 5:'Friday', 6:'Saturday', 0:'Sunday'}
def weekday(l):
    c = int(l[0])*10+int(l[1])
    y = int(l[2])*10+int(l[3])
    if (l[5]=='1' or l[5]=='2') and l[4]=='0':
        m = int(l[5]) + 12
        if y == 0 :
            c -= 1
            y = 99
        else:
            y -= 1
    else:
        m = int(l[5])+int(l[4])*10
        d = int(l[6])*10+int(l[7])
        w = (y+math.floor(y/4)+math.floor(c/4)-2*c+math.floor(26*(m+1)/10)+d-1)%7
    return w
n = int(input())
for _ in range(n):
    l = list(input())
    print(num_weekday[weekday(l)])

```

基本信息

#: 50198443
 题目: M19944
 提交人: Rose-Rong
 内存: 3660kB
 时间: 21ms
 语言: Python3
 提交时间: 2025-09-30 18:06:06

```

import math
num_weekday =
{1:'Monday', 2:'Tuesday', 3:'Wednesday', 4:'Thursday', 5:'Friday', 6:'Saturday', 0:'Sunday'}
def weekday(l):
    c = int(l[0])*10+int(l[1])
    y = int(l[2])*10+int(l[3])
    if (l[5]=='1' or l[5]=='2') and l[4]=='0':
        m = int(l[5]) + 12
        if y == 0 :
            c -= 1
            y = 99
        else:
            y -= 1
    else:
        m = int(l[5])+int(l[4])*10
        d = int(l[6])*10+int(l[7])
        w = (y+math.floor(y/4)+math.floor(c/4)-2*c+math.floor(26*(m+1)/10)+d-1)%7
    return w
n = int(input())
for _ in range(n):

```

```
l = list(input())
print(num_weekday[weekday(l)])
```

10月5-7日笔记

1475A. Odd Divisor

By Rose-Rong, contest: Codeforces Round 697 (Div. 3), problem: (A) Odd Divisor, Accepted, #, [Copy](#)

```
t = int(input())
for _ in range(t):
    num = int(input())
    if num%2 == 0:
        while True:
            if num%2 != 0:
                if num==1:
                    print('NO')
                else:
                    print('YES')
                break
            else:
                num = num//2
    else:
        if num!=1:
            print('YES')
        else:
            print('NO')
```

01002: 方便记忆的电话号码

Version 1

```
n = int(input())
set_num = set()
ans = []
dict_num =
{'A':'2', 'B':'2', 'C':'2', 'D':'3', 'E':'3', 'F':'3', 'G':'4', 'H':'4',
'I':'4', 'J':'5', 'K':'5', 'L':'5', 'M':'6', 'N':'6', 'O':'6', 'P':'7',
'R':'7', 'S':'7', 'T':'8', 'U':'8', 'V':'8', 'W':'9', 'X':'9', 'Y':'9'}
}
for _ in range(n):
    new = input()
    new_num = []
    for i in range(len(new)):
        if new[i] in dict_num.values() or new[i]=='1' or
```

```

new[i] = '0':
    new_num.append(new[i])
    if new[i] in dict_num.keys():
        new_num.append(dict_num[new[i]])
new_num = ''.join(new_num)
formatted_num = new_num[:3] + '-' + new_num[3:]
set_num.add(formatted_num)
ans.append(formatted_num)

sorted_set = sorted(set_num)
res = 0
for j in sorted_set:
    cnt = ans.count(j)
    if cnt > 1:
        print(f'{j} {cnt}')
        res += 1
if res == 0:
    print('No duplicates')

```

这是自己手搓的代码，使用集合去重以计数，使用ans[]来存储所有重排好的电话号码。

Time Limit Exceeded

Reasons:

1. 低效的计数方法

使用count来计数，若遇到test_cases极多且'No Duplicates'的情况，仍需要每次从集合的第一个开始遍历，当然会超时。

2. if new[i] in dict_num.values() or new[i] == 'i' or new[i] == 'o':

这里很繁琐，直接用.isdigit/.isalpha/in '0123456789'即可

3. formatted_num = new_num[:3] + '-' + new_num[3:]

这一步还想了很久



状态: Accepted

源代码

```

n = int(input())
dict_num = {'A': '2', 'B': '2', 'C': '2', 'D': '3', 'E': '3', 'F': '3', 'G': '4', 'H': '4', 'I': '4', 'J': '5', 'K': '5', 'L': '5', 'M': '6', 'N': '6', 'O': '6', 'P': '7', 'R': '7', 'S': '7', 'T': '8', 'U': '8', 'V': '8', 'W': '9', 'X': '9', 'Y': '9'}
cnt_dict = {}
for _ in range(n):
    new_phos = []
    new = input()
    for char in new:
        if char == '-':
            continue
        if char.isdigit():
            new_phos.append(char)
        elif char.isalpha():
            new_phos.append(dict_num[char.upper()])
    new_phos = ''.join(new_phos)
    formatted_phos = new_phos[:3] + '-' + new_phos[3:]
    if formatted_phos in cnt_dict:
        cnt_dict[formatted_phos] += 1
    else:
        cnt_dict[formatted_phos] = 1

sorted_dict = sorted(cnt_dict)
has_duplicates = False
for number in sorted_dict:
    cnt = cnt_dict[number]
    if cnt > 1:
        has_duplicates = True
        print(f'{number} {cnt}')

if not has_duplicates:
    print('No duplicates.')

```

基本信息

#: 50240163
 题目: M01002
 提交人: Rose-Rong
 内存: 13168kB
 时间: 510ms
 语言: Python3
 提交时间: 2025-10-06 17:09:31

```

n = int(input())
dict_num =
{'A': '2', 'B': '2', 'C': '2', 'D': '3', 'E': '3', 'F': '3', 'G': '4', 'H': '4', 'I': '4', 'J': '5', 'K': '5', 'L': '5', 'M': '6', 'N': '6', 'O': '6', 'P': '7', 'R': '7', 'S': '7', 'T': '8', 'U': '8', 'V': '8', 'W': '9', 'X': '9', 'Y': '9'}
cnt_dict = {}
for _ in range(n):
    new_phos = []
    new = input()
    for char in new:
        if char == '-':
            continue
        if char.isdigit():
            new_phos.append(char)
        elif char.isalpha():
            new_phos.append(dict_num[char.upper()]))

```

```

new_pho = ''.join(new_pho)
formatted_pho = new_pho[:3] + '-' + new_pho[3:]
if formatted_pho in cnt_dict:
    cnt_dict[formatted_pho] += 1
else:
    cnt_dict[formatted_pho] = 1

sorted_dict = sorted(cnt_dict)
has_duplicates = False
for number in sorted_dict:
    cnt = cnt_dict[number]
    if cnt > 1:
        has_duplicates = True
        print(f'{number} {cnt}')

if not has_duplicates:
    print('No duplicates.')

```

1. 使用字典计数时间复杂度会小很多！

可以起到遇到一个就计数一个的效果，不用从头遍历！而且可以自动添加元素！

OJ不允许import操作

(其实from collections import Counter也可)

cnt_dict = Counter()

cnt_dict[formatted_pho] += 1

手搓判断操作：

if formatted_pho in cnt_dict:

cnt_dict[formatted_pho] += 1

else:

cnt_dict[formatted_pho] = 1

2. 判断'No Duplicates'的操作优化：

has_duplicates = False(先定义has_duplicates是False,后来看是否修改)

for number in sorted_dict:

cnt = cnt_dict[number]

if cnt > 1:

has_duplicates = True

print(f'{number} {cnt}')

```
if not has_duplicates:  
    print('No duplicates.') (句点不要忘记了! )
```

04015: 邮箱验证

#50241254提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
while True:  
    try:  
        s = input()  
    except EOFError:  
        break  
  
    if s.count('@') != 1:  
        print('NO')  
        continue  
    if s[0] in ['@', '.'] or s[-1] in ['@', '.']:  
        print('NO')  
        continue  
    if s[s.index('@')+1] == '.' or s[s.index('@)-1] == '.':  
        print('NO')  
        continue  
    if '.' not in s[s.index('@')+2:]:  
        print('NO')  
        continue  
    print('YES')
```

基本信息

#: 50241254
题目: M04015
提交人: Rose-Rong
内存: 3592kB
时间: 20ms
语言: Python3
提交时间: 2025-10-06 20:11:12

```
while True:  
    try:  
        s = input()  
    except EOFError:  
        break  
  
    if s.count('@')!=1:  
        print('NO')  
        continue  
    if s[0] in ['@', '.'] or s[-1] in ['@', '.']:  
        print('NO')  
        continue  
    if s[s.index('@')+1]== '.' or s[s.index('@)-1]== '.':  
        print('NO')  
        continue  
    if '.' not in s[s.index('@')+2:]:  
        print('NO')  
        continue  
    print('YES')
```

'@'不能和'.'直接相连→会有@.&@两种情况!

s[0] in ['@','.] or s[-1] in ['@','.]

02910: 提取数字

#50241803提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
s = input()
ans = []
for i in range(len(s)):
    if ans=='0' and s[i].isdigit():
        ans.clear()
    if s[i].isdigit():
        ans.append(s[i])
    else:
        if ans!=[]:
            print(''.join(ans))
            ans.clear()
        if i+1==len(s):
            break
if ans!=[]:
    print(''.join(ans))
```

基本信息

#: 50241803
题目: E02910
提交人: Rose-Rong
内存: 3608kB
时间: 19ms
语言: Python3
提交时间: 2025-10-06 21:35:31

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

先导0一开始没考虑

input sample: ao123boooocod

output sample: 123 o o

96A. Football

By Rose-Rong, contest: Codeforces Beta Round 77 (Div. 2 Only), problem: (A) Football, Accepted, #, Copy

```
s = input()
ans = 1
dangerous = False
for i in range(len(s)):
    if i + 1 == len(s):
        break
    if s[i]==s[i+1]:
        ans+=1
    else:
        ans=1
    if ans==7:
        dangerous = True
        break
print('YES' if dangerous else 'NO')
```

RE的几种错误:

- 1.除以0错误
- 2.数组/列表索引越界
- 3.空指针/空引用
- 4.类型错误

5.内存不足

当程序使用过多内存时发生

6.递归深度过大

7.文件/输入输出错误

20742: 泰波拿契數

#50242304提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
n = int(input())
s = [0,1,1]
if n==1 or n==2:
    print(1)
if n>2:
    for i in range(3,n+1):
        t_new=int(s[i-3])+int(s[i-2])+int(s[i-1])
        s.append(t_new)
    print(t_new)
```

基本信息

#: 50242304
题目: E20742
提交人: Rose-Rong
内存: 3608kB
时间: 21ms
语言: Python3
提交时间: 2025-10-06 23:13:15

©2002-2022 POJ 京ICP备20010980号-1

English 帮助 关于

58A. Chat room

By Rose-Rong, contest: Codeforces Beta Round 54 (Div. 2), problem: (A) Chat room, Accepted, #, [Copy](#)

```
s = input()
standard_list = ['h', 'e', 'l', 'l', 'o']
alphabet_list = []
j = 0
for i in range(len(s)):
    if s[i] == standard_list[j]:
        alphabet_list.append(s[i])
        j+=1
    if j == 5:
        break
formatted_word = ''.join(alphabet_list)
if formatted_word=='hello':
    print('YES')
else:
    print('NO')
```

02786: Pell数列

程序应该没问题，但是超时，容我再想想。