Assignment #2: 字符串相关

Updated 2211 GMT+8 Sep 19, 2023

2023 fall, Complied by Hongfei Yan==（请改为同学的姓名、院系）==

Markdown（用 https://typoraio.cn 编辑）格式文件在，https://github.com/GMyhf/2023fall-cs101

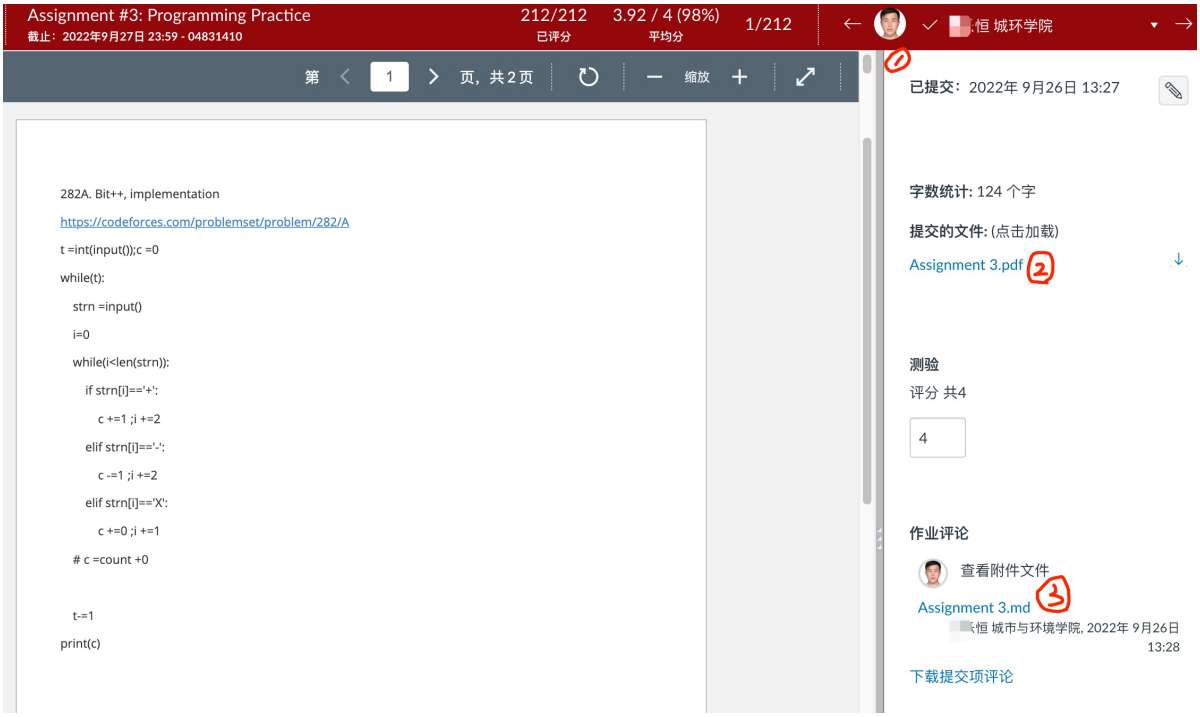
课程号: 04831410 课程名: 计算概论(B)	**班号: 12**
上课时间: 1-16周 每周 周二 7-9节	地点: 理教208
上机时间: 2-15周 每周 周四 7-8节 期末机考时间: 第16周 周四 7-8节	地点：理科1号楼计算中心，二层楼的6号和三层楼的7号机房
助教：张哲瑞、张以宁、彭亦男、涂程颖、陈威宇	在课程微信群中的名字是"TA-"开始，地点：理科1号楼1220

说明：

1）第2周课上讲到了计算机相关的历史，介绍了ASCII表。

2）请把每个题目解题思路（可选），源码Python, 或者C++/C（已经在Codeforces/Openjudge上AC），截图（包含Accepted, 学号，或者姓名+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"。提交后，助教看到画面如下，有同学清晰头像、提交文件有pdf、作业评论有md或者doc。



4）同学完成作业的时候，就是这个模版文件中修改补充好。为便于助教批改作业，请尽量不要删除其他文字。

5）如果不能在截止前提交作业，请写明原因。

编程环境

==（请改为同学的操作系统、编程环境等）==

操作系统：Win11

Python编程环境：Spyder IDE 5.2.2

1. 必做题目

71A. Way Too Long Words

strings, 1000, http://codeforces.com/problemset/problem/71/A

Sometimes some words like "*localization*" or "*internationalization*" are so long that writing them many times in one text is quite tiresome.

Let's consider a word *too long*, if its length is **strictly more** than 10 characters. All too long words should be replaced with a special abbreviation.

This abbreviation is made like this: we write down the first and the last letter of a word and between them we write the number of letters between the first and the last letters. That number is in decimal system and doesn't contain any leading zeroes.

Thus, "*localization*" will be spelt as "*l10n*", and "*internationalization*" will be spelt as "i18n".

You are suggested to automatize the process of changing the words with abbreviations. At that all too long words should be replaced by the abbreviation and the words that are not too long should not undergo any changes.

Input

The first line contains an integer n ($1 \le n \le 100$). Each of the following n lines contains one word. All the words consist of lowercase Latin letters and possess the lengths of from 1 to 100 characters.

Output

Print n lines. The i-th line should contain the result of replacing of the i-th word from the input data.

Examples

input

```
4
word
localization
internationalization
pneumonoultramicroscopicsilicovolcanoconiosis
```

output

```
word
l10n
i18n
p43s
```

【苏王捷，工学院，2023年秋】 ==（请改为同学的姓名、院系）==

思路：判断是否需要操作后保留前后添加数字

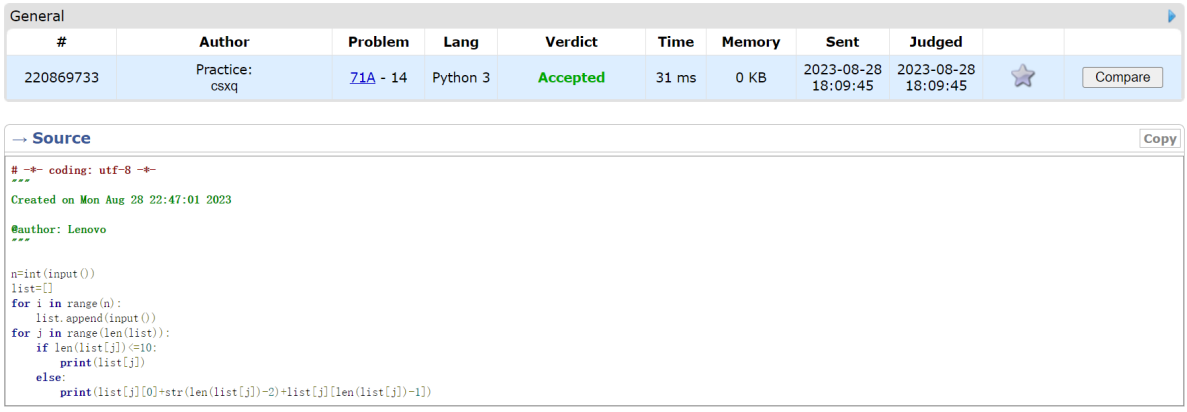
Python3 代码

```python
# -*- coding: utf-8 -*-
"""
Created on Mon Aug 28 22:47:01 2023

@author: 苏王捷 2300011075
"""

n=int(input())
list=[]
for i in range(n):
    list.append(input())
for j in range(len(list)):
    if len(list[j])<=10:
        print(list[j])
    else:
        print(list[j][0]+str(len(list[j])-2)+list[j][len(list[j])-1])
```

Python代码运行截图 ==（请替换为同学的AC代码截图，至少包含有"Accepted"，和“学号”的截图）==



C++ 代码（如果没有，C++部分可以删除）

```cpp
// 请改为同学的代码
```

C++ 代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

112A. Petya and Strings

implementation/strings, 800, http://codeforces.com/problemset/problem/112/A

Little Petya loves presents. His mum bought him two strings of the same size for his birthday. The strings consist of uppercase and lowercase Latin letters. Now Petya wants to compare those two strings lexicographically. The letters' case does not matter, that is an uppercase letter is considered equivalent to the corresponding lowercase letter. Help Petya perform the comparison.

Input

Each of the first two lines contains a bought string. The strings' lengths range from 1 to 100 inclusive. It is guaranteed that the strings are of the same length and also consist of uppercase and lowercase Latin letters.

Output

If the first string is less than the second one, print "-1". If the second string is less than the first one, print "1". If the strings are equal, print "0". Note that the letters' case is not taken into consideration when the strings are compared.

Examples

input

```
aaaa
aaaA
```

output

```
0
```

input

```
abs
Abz
```

output

```
-1
```

input

```
abcdefg
AbCdEfF
```

output

```
1
```

Note

If you want more formal information about the lexicographical order (also known as the "dictionary order" or "alphabetical order"), you can visit the following site:

- http://en.wikipedia.org/wiki/Lexicographical_order

【苏王捷，工学院，2023年秋】 ==（请改为同学的姓名、院系）==

思路：大小写统一后直接比较

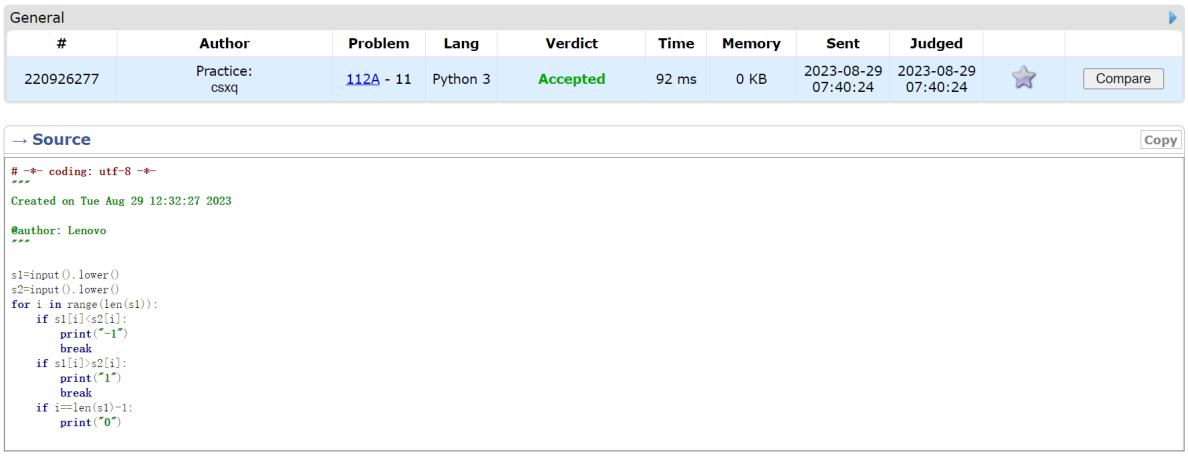
Python3 代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Aug 29 12:32:27 2023

@author: 苏王捷 2300011075
"""

s1=input().lower()
s2=input().lower()
for i in range(len(s1)):
    if s1[i]<s2[i]:
        print("-1")
        break
    if s1[i]>s2[i]:
        print("1")
        break
    if i==len(s1)-1:
        print("0")
```

Python代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==



C++ 代码（如果没有，C++部分可以删除）

```
// 请改为同学的代码

```

C++ 代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

158A. Next Round

*special problem/implementation, 800, http://codeforces.com/problemset/problem/158/A

"Contestant who earns a score equal to or greater than the *k*-th place finisher's score will advance to the next round, as long as the contestant earns a positive score..." — an excerpt from contest rules.

A total of *n* participants took part in the contest ($n \geq k$), and you already know their scores. Calculate how many participants will advance to the next round.

Input

The first line of the input contains two integers *n* and *k* ($1 \leq k \leq n \leq 50$) separated by a single space.

The second line contains *n* space-separated integers *a*~1~, *a*~2~, ..., a~n~ ($0 \leq a_i \leq 100$), where a~i~ is the score earned by the participant who got the *i*-th place. The given sequence is non-increasing (that is, for all *i* from 1 to *n* - 1 the following condition is fulfilled: $a_i \geq a_i + 1$).

Output

Output the number of participants who advance to the next round.

Examples

input

```
8 5
10 9 8 7 7 7 5 5
```

output

```
6
```

input

```
4 2
0 0 0 0
```

output

```
0
```

Note

In the first example the participant on the 5th place earned 7 points. As the participant on the 6th place also earned 7 points, there are 6 advancers.

In the second example nobody got a positive score.

【苏王捷，工学院，2023年秋】 ==（请改为同学的姓名、院系）==

思路：取原先边界值再将后面可能进入下一轮的人加入

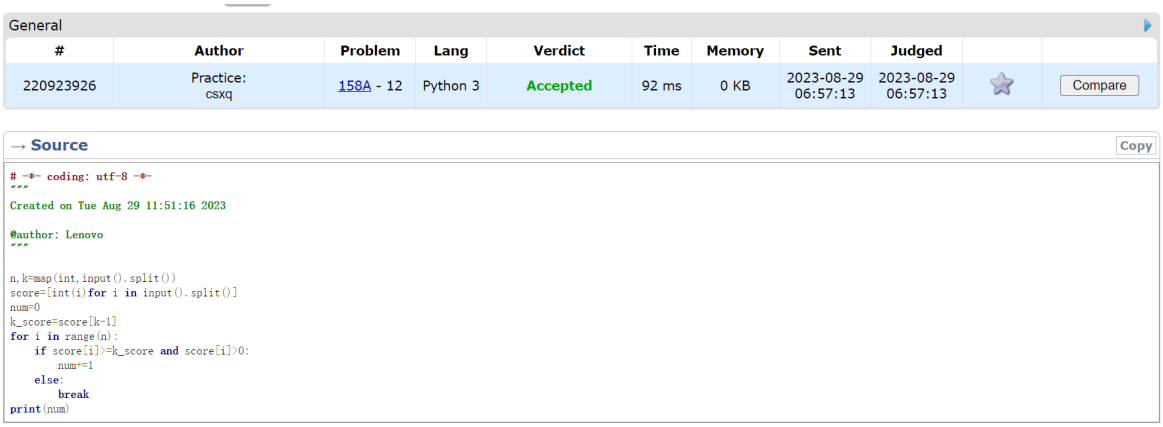
Python3 代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Aug 29 11:51:16 2023

@author: 苏王捷 2300011075
"""

n,k=map(int,input().split())
score=[int(i)for i in input().split()]
num=0
k_score=score[k-1]
for i in range(n):
    if score[i]>=k_score and score[i]>0:
        num+=1
    else:
        break
print(num)
```

Python代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==



C++ 代码（如果没有，C++部分可以删除）

```
// 请改为同学的代码
```

C++ 代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

58A. Chat room

greedy/strings, 1000, http://codeforces.com/problemset/problem/58/A

Vasya has recently learned to type and log on to the Internet. He immediately entered a chat room and decided to say hello to everybody. Vasya typed the word *s*. It is considered that Vasya managed to say hello if several letters can be deleted from the typed word so that it resulted in the word "hello". For example, if Vasya types the word "ahhellllloou", it will be considered that he said hello, and if he types "hlelo", it will be considered that Vasya got misunderstood and he didn't manage to say hello. Determine whether Vasya managed to say hello by the given word *s*.

Input

The first and only line contains the word *s*, which Vasya typed. This word consisits of small Latin letters, its length is no less that 1 and no more than 100 letters.

Output

If Vasya managed to say hello, print "YES", otherwise print "NO".

Examples

input

```
ahhellllloou
```

output

```
YES
```

input

```
hlelo
```

output

```
NO
```

【苏王捷，工学院，2023年秋】 ==（请改为同学的姓名、院系）==

思路：将hello储存，出现可能的hello时逐一添加，最后判断是否成立

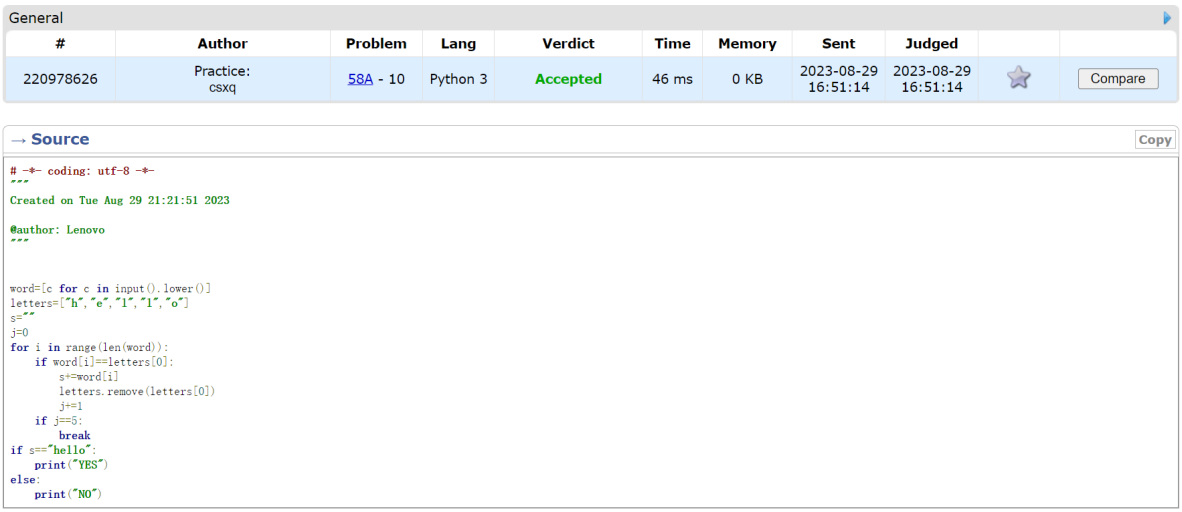
Python3 代码

```python
# -*- coding: utf-8 -*-
"""
Created on Tue Aug 29 21:21:51 2023

@author:苏王捷 2300011075
"""


word=[c for c in input().lower()]
letters=["h","e","l","l","o"]
s=""
j=0
for i in range(len(word)):
    if word[i]==letters[0]:
        s+=word[i]
        letters.remove(letters[0])
        j+=1
    if j==5:
        break
if s=="hello":
    print("YES")
else:
    print("NO")
```

Python代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==



C++ 代码（如果没有，C++部分可以删除）

```cpp
// 请改为同学的代码
```

C++ 代码运行截图 ==（AC代码截图，至少包含有"Accepted"）==

04015: 邮箱验证

strings, http://cs101.openjudge.cn/practice/04015

POJ 注册的时候需要用户输入邮箱，验证邮箱的规则包括：
1)有且仅有一个'@'符号
2)'@'和'.'不能出现在字符串的首和尾
3)'@'之后至少要有一个'.'，并且'@'不能和'.'直接相连
满足以上3条的字符串为合法邮箱，否则不合法，
编写程序验证输入是否合法

输入

输入包含若干行，每一行为一个代验证的邮箱地址，长度小于100

输出

每一行输入对应一行输出
如果验证合法，输出 YES
如果验证非法：输出 NO

样例输入

```
.a@b.com
pku@edu.cn
cs101@gmail.com
cs101@gmail
```

样例输出

```
NO
YES
YES
NO
```

【苏王捷，工学院，2023年秋】 ==（请改为同学的姓名、院系）==

思路：以.和@入手判断

Python3 代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Sep 19 23:29:13 2023

@author: 苏王捷 2300011075
```

```
"""

def check(email):
    if len(email)!=2:
        return "NO"
    elif email[0]=="" or email[1]=="":
        return "NO"
    elif email[0][0]=="." or email[0][-1]==".":
        return "NO"
    else:
        last=sorted(email[1].split("."))
        if last[0]=="" or len(last)==1:
            return "NO"
        else:
            return "YES"
l=[]
while True:
    try:
        email=input().split("@")
        l.append(check(email))
    except:
        break
for i in l:
    print(i)
```

Python代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

#41258704提交状态

查看　提交　统计　提问

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Tue Sep 19 23:29:13 2023

@author: Lenovo
"""

def check(email):
    if len(email)!=2:
        return "NO"
    elif email[0]=="" or email[1]=="":
        return "NO"
    elif email[0][0]="." or email[0][-1]=".":
        return "NO"
    else:
        last=sorted(email[1].split("."))
        if last[0]=="" or len(last)==1:
            return "NO"
        else:
            return "YES"
l=[]
while True:
    try:
        email=input().split("@")
        l.append(check(email))
    except:
        break
for i in l:
    print(i)
```

基本信息

#:	41258704
题目:	04015
提交人:	23n2300011075
内存:	3544kB
时间:	30ms
语言:	Python3
提交时间:	2023-09-20 00:43:13

C++ 代码（如果没有，C++部分可以删除）

```
// 请改为同学的代码

```

C++ 代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

2. 选做题目

OJ01008: Maya Calendar

math, http://cs101.openjudge.cn/practice/01008/

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 then 5000.

输出

The date in Tzolkin should be in the following format:
Number NameOfTheDay Year

The first line of the output file contains the number of the output dates. In the next n lines, there are dates in the Tzolkin calendar format, in the order corresponding to the input dates.

样例输入

```
3
10. zac 0
0. pop 0
10. zac 1995
```

样例输出

```
3
3 chuen 0
1 imix 0
9 cimi 2801
```

来源

Central Europe 1995

【苏王捷，工学院，2023年秋】 ==（请改为同学的姓名、院系）==

思路：字典取序，取余后相乘

Python3 代码

```
n=int(input())
days=[]
for i in range(n):
    day=input()
    d=day.replace(".","")
    days.append(list(d.split()))
dic_Haab=
{"pop":0,"no":1,"zip":2,"zotz":3,"tzec":4,"xul":5,"yoxkin":6,"mol":7,"chen":8,"ya
x":9,"zac":10,"ceh":11,"mac":12,"kankin":13,"muan":14,"pax":15,"koyab":16,"cumhu"
:17,"uayet":18}
dic_Tzolkin=
{1:"imix",2:"ik",3:"akbal",4:"kan",5:"chicchan",6:"cimi",7:"manik",8:"lamat",9:"m
uluk",10:"ok",11:"chuen",12:"eb",13:"ben",14:"ix",15:"mem",16:"cib",17:"caban",18
:"eznab",19:"canac",20:"ahau"}
day_new=[n]
for i in days:
    date=int(i[0])+dic_Haab[i[1]]*20+int(i[2])*365
    year=date//260
    month=dic_Tzolkin[date%260%20+1]
    day=date%260%13+1
    date=str(day)+" "+month+" "+str(year)
```

```
    day_new.append(date)
for i in range(n+1):
    print(day_new[i])
```

Python代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

C++ 代码（如果没有，C++部分可以删除）

```
// 请改为同学的代码
```

C++ 代码运行截图 ==（AC代码截图，至少包含有"Accepted"，和“学号”）==

3. 学习总结和收获

每天两到三个oj新题，也在尝试使用C++()

#41251720提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
l=[]
A=True
while A:
    days=input()
    if days!="-1 -1 -1 -1":
        l.append(list(map(int,days.split())))
    else:
        A=False
n=0
for queue in l:
    n+=1;B=True
    q,e,i,d=map(int,queue)
    q=q%23;e=e%28;i=i%33
    add=23-q
    while B:
        if (e+add)%28==0 and (i+add)%33==0:
            B=False
        else:
            add+=23
    d+=add
    if d>=21252:
        num=42504-d
    else:
        num=21252-d
    print(f'Case {n}: the next triple peak occurs in {num} days.')
```

基本信息

#: 41251720
题目: 01006
提交人: 23n2300011075
内存: 34436kB
时间: 100ms
语言: PyPy3
提交时间: 2023-09-19 16:25:52

#41252560提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
number=[]
while True:
    try:
        number.append(str(input()))
    except:
        break
for num in number:
    if str(int(num)*(len(num)+1))=="9"*len(num):
        print(f'{num} is cyclic')
    else:
        print(f'{num} is not cyclic')
```

基本信息

#: 41252560
题目: 01047
提交人: 23n2300011075
内存: 3672kB
时间: 28ms
语言: Python3
提交时间: 2023-09-19 17:13:47

#41242131提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
try:
    while True:
        r, n = map(float, input().split())
        n = int(n)
        r_without_point = r

        if r % 1 != 0:
            last_location = len(str(r).split('.')[-1])
        else:
            last_location = 0

        r_without_point *= 10**last_location

        r_without_point = int(r_without_point)
        output = str(r_without_point**n)
        if r < 1:
            print(f'.{str(0)*(last_location*n - len(output))}{output}')
        elif last_location == 0:
            print(int(r ** n))
        elif r > 1:
            print(f'{output[:len(output) - last_location*n]}.{output[len(output) -

except EOFError:
    pass
```

基本信息

#: 41242131
题目: 01001
提交人: 23n2300011075
内存: 3736kB
时间: 27ms
语言: Python3
提交时间: 2023-09-18 19:06:52

#41237537提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
import math
n=int(input())
l=[]
for i in range(n):
    p=int(input())
    total,add=1,1
    i=1
    while total<p:
        i+=1
        add=add+int(math.log10(i))+1
        total+=add
    total-=add
    pos=p-total
    i=1
    while pos>=(int(math.log10(i))+1):
        pos-=int(math.log10(i))+1
        i+=1
    if pos==0:
        digit=int(str(i-1)[-1])
    else:
        digit=int(str(i)[pos-1])
    l.append(digit)
for i in l:
    print(i)
```

基本信息

#: 41237537
题目: 01019
提交人: 23n2300011075
内存: 3736kB
时间: 115ms
语言: Python3
提交时间: 2023-09-18 15:33:17

#41221795提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
# -*- coding: utf-8 -*-
"""
Created on Sun Sep 17 14:09:23 2023

@author: Lenovo
"""

a=True
list=[]
while a:
    goal=float(input())
    if goal==0.0:
        a=False
    else:
        n,length=0,0
        while length<goal:
            n+=1
            length+=1/(n+1)
        list.append(n)
for n in list:
    print(f"{n} card(s)")
```

基本信息

#: 41221795
题目: 01003
提交人: 23n2300011075
内存: 3584kB
时间: 29ms
语言: Python3
提交时间: 2023-09-17 14:29:28

附录

如果设好了 typora的图床，md文件中图片在其他地方也能看见。因为图片存在云端。如果不好设置，注意导出的pdf文件包含图片。

Typora+PicGo+Github解决个人博客图片上传问题 https://zhuanlan.zhihu.com/p/367529569