

[Xâu ký tự]. Bài 1. Đếm số loại ký tự trong xâu.

Cho một xâu ký tự, hay đếm số lượng ký tự là chữ cái, chữ số và ký tự đặc biệt (Các ký tự không phải là chữ cái và chữ số).

Input Format

Xâu ký tự S.

Constraints

$1 \leq \text{len}(S) \leq 1000$.

Output Format

In ra số lượng chữ cái, chữ số, và ký tự đặc biệt xuất hiện trong xâu

Sample Input 0

```
abcd123 $%^ a
```

Sample Output 0

```
5 3 5
```

[Xâu ký tự]. Bài 2. In hoa, in thường.

Cho một xâu ký tự, hãy viết hoa và viết thường các ký tự là chữ cái trong xâu, các ký tự không phải là chữ cái thì không thay đổi.

Input Format

Xâu ký tự S.

Constraints

$1 \leq \text{len}(S) \leq 1000$.

Output Format

Dòng đầu tiên in ra xâu sau khi thay đổi mọi ký tự thường trong xâu thành ký tự hoa. Dòng thứ 2 in ra xâu sau khi thay đổi mọi ký tự hoa trong xâu thành ký tự thường.

Sample Input 0

```
abcd ABCD
```

Sample Output 0

```
ABCD ABCD  
abcd abcd
```

[Xâu ký tự]. Bài 3. Tần suất xuất hiện của ký tự

Cho một xâu kí tự, hãy đếm tần suất xuất hiện của các kí tự trong xâu và in ra theo yêu cầu.

Input Format

Xâu kí tự S chỉ bao gồm chữ cái in hoa và in thường.

Constraints

$1 \leq \text{len}(S) \leq 100000$.

Output Format

Đầu tiên in ra các ký tự và tần suất xuất hiện của các ký tự ở trong xâu theo thứ tự từ điển tăng dần, sau đó cách ra một dòng và in ra tần suất xuất hiện của các ký tự theo thứ tự xuất hiện trong xâu(chú ý mỗi ký tự chỉ in 1 lần)

Sample Input 0

```
abcdabcdA
```

Sample Output 0

```
A 1  
a 2  
b 2  
c 2  
d 2  
  
a 2  
b 2  
c 2  
d 2  
A 1
```

[Xâu ký tự]. Bài 4. Ký tự xuất hiện nhiều nhất trong xâu.

Cho một chuỗi ký tự, hãy tìm ký tự có số lần xuất hiện ít nhất trong chuỗi và ký tự có số lần xuất hiện nhiều nhất ở trong chuỗi. Trong trường hợp có nhiều ký tự có cùng số lần xuất hiện lớn nhất hoặc nhỏ nhất thì in ra ký tự có thứ tự từ điển lớn nhất.

Input Format

Chuỗi ký tự S chỉ bao gồm chữ cái in hoa và in thường.

Constraints

$1 \leq \text{len}(S) \leq 100000$.

Output Format

Dòng đầu tiên in ra ký tự có số lần xuất hiện nhiều nhất, trong trường hợp có nhiều ký tự có cùng số lần xuất hiện nhiều nhất thì chọn ký tự có thứ tự từ điển lớn nhất. Dòng thứ 2 in ra ký tự có số lần xuất hiện nhỏ nhất, trong trường hợp có nhiều ký tự có cùng số lần xuất hiện nhỏ nhất thì chọn ký tự có thứ tự từ điển lớn nhất.

Sample Input 0

```
qEQtEEuAQcSWfFRfWRnKKeCZeUYlSGrXWvOGqEXsWMxLRtQPxBMtEHdPGbGKvKFjQLxU  
SuSZtMIjXEHUZtAZdBInYLeHWfWUwCVo
```

Sample Output 0

```
W 6  
w 1
```

[Xâu ký tự]. Bài 5. Ký tự xuất hiện ở cả 2 chuỗi.

Cho 2 chuỗi ký tự S1 và S2, hãy in ra các ký tự xuất hiện ở cả 2 chuỗi theo thứ tự từ điển, chú ý mỗi ký tự chỉ liệt kê một lần. Sau đó tiếp tục liệt kê các ký tự xuất hiện ở 1 trong 2 chuỗi theo thứ tự từ điển.

Input Format

Dòng đầu tiên là chuỗi S1. Dòng thứ 2 là chuỗi S2. Các ký tự trong 2 chuỗi chỉ bao gồm chữ cái in hoa hoặc in thường.

Constraints

$1 \leq \text{len}(S1) \leq 100000$; $1 \leq \text{len}(S2) \leq 100000$.

Output Format

Dòng 1 in ra các ký tự xuất hiện ở cả 2 xâu theo thứ tự từ điển tăng dần. Dòng 2 in ra các ký tự xuất hiện ở 1 trong 2 xâu theo thứ tự từ điển tăng dần.

Sample Input 0

```
nTOhOGvRBk  
bUJtZSoONh
```

Sample Output 0

```
Oh  
BGJNORSTUZbhknotv
```

[Xâu ký tự]. Bài 6. Ký tự xuất hiện ở 2 xâu 2.

Cho 2 xâu ký tự S1 và S2 chỉ bao gồm chữ cái in hoa và in thường, hãy tìm các ký tự xuất hiện trong xâu S1 mà không xuất hiện trong xâu S2, và các ký tự chỉ xuất hiện trong xâu S2 mà không xuất hiện trong xâu S1. Các ký tự được in ra theo thứ tự từ điển và chỉ liệt kê mỗi ký tự một lần.

Input Format

Dòng đầu tiên là xâu S1. Dòng thứ 2 là xâu S2.

Constraints

$1 \leq \text{len}(S1) \leq 100000$; $1 \leq \text{len}(S2) \leq 100000$.

Output Format

Dòng đầu tiên in ra các ký tự chỉ xuất hiện trong S1 mà không xuất hiện trong S2. Dòng thứ 2 in ra các ký tự chỉ xuất hiện trong S2 mà không xuất hiện trong S1.

Sample Input 0

```
fAOxTUEaIs  
aVUkULeRYw
```

Sample Output 0

```
AIOTfsx  
LRVYakw
```

[Xâu ký tự]. Bài 7. Xâu đối xứng.

Cho một xâu ký tự S chỉ bao gồm các chữ cái, hãy kiểm tra xem xâu nhập vào có phải là xâu đối xứng hay không?

Input Format

Dòng duy nhất là xâu S

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra YES nếu S đối xứng, ngược lại in NO.

Sample Input 0

```
cBGkXXqKWqQWKqXXkGBc
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 8. Xâu pangram 1

Xâu pangram là xâu có chứa đầy đủ các ký tự từ A tới Z không phân biệt chữ hoa hay thường. Nhập vào xâu S và kiểm tra xem xâu S có phải là xâu pangram hay không?

Input Format

Xâu ký tự S chỉ bao gồm các ký tự in hoa hoặc in thường.

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra YES nếu S là xâu pangram, ngược lại in NO.

Sample Input 0

```
abcdefghijklmnopqrstuvwxyz
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 9. Đếm số lượng từ trong xâu

Cho một chuỗi ký tự S bao gồm các chữ cái và dấu cách, một từ được định nghĩa là các ký tự liên tiếp không chứa dấu cách, hãy đếm số lượng từ xuất hiện trong chuỗi S .

Input Format

Dòng duy nhất chứa chuỗi S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra số lượng từ xuất hiện trong chuỗi S .

Sample Input 0

```
ngon ngu lap trinh java, python
```

Sample Output 0

```
6
```

[Xâu ký tự]. Bài 10. Liệt kê các từ khác nhau trong xâu.

Cho một chuỗi ký tự S bao gồm các chữ cái và dấu cách, một từ được định nghĩa là các ký tự liên tiếp không chứa dấu cách, hãy liệt kê các từ khác nhau trong chuỗi S , đầu tiên hãy liệt kê các từ khác nhau theo thứ tự từ điển tăng dần, sau đó liệt kê các từ theo thứ tự xuất hiện trong xâu.

Input Format

Dòng duy nhất chứa chuỗi S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

Dòng đầu tiên in ra các từ trong xâu theo thứ tự từ điển. Dòng thứ hai in ra các từ theo thứ tự xuất hiện trong xâu. Chú ý không in dấu cách thừa sau từ cuối cùng của từng dòng.

Sample Input 0

```
python java php php java python
```

Sample Output 0

```
java php python
python java php
```

[Xâu ký tự]. Bài 11. Sắp xếp các từ trong xâu 1

Cho một xâu ký tự S bao gồm các chữ cái và dấu cách, một từ được định nghĩa là các ký tự liên tiếp không chứa dấu cách, đầu tiên hãy sắp xếp các từ trong xâu theo thứ tự từ điển tăng dần, sau đó sắp xếp các từ trong xâu theo thứ tự chiều dài tăng dần, trong trường hợp có nhiều từ có cùng chiều dài thì từ nào có thứ tự từ điển nhỏ hơn sẽ in ra trước.

Input Format

Dòng duy nhất chứa xâu S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

Dòng đầu tiên in ra các từ trong xâu theo thứ tự từ điển tăng dần. Dòng thứ 2 in ra các từ trong xâu theo thứ tự chiều dài tăng dần, trong trường hợp có nhiều từ có cùng chiều dài thì từ nào có thứ tự từ điển nhỏ hơn sẽ in ra trước. Các từ được in cách nhau một dấu cách.

Sample Input 0

```
hoc lap trinh php
```

Sample Output 0

```
hoc lap php trinh
hoc lap php trinh
```

[Xâu ký tự]. Bài 12. Sắp xếp các từ trong xâu 2

Cho một xâu ký tự S bao gồm các chữ cái và dấu cách, một từ được định nghĩa là các ký tự liên tiếp không chứa dấu cách, hãy sắp xếp các từ thuận nghịch khác nhau trong xâu theo thứ tự từ chiều dài tăng dần, nếu 2 từ thuận nghịch có cùng chiều dài thì từ nào xuất hiện trước sẽ được in ra trước.

Input Format

Dòng duy nhất chứa xâu S.

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra các từ theo thứ tự sắp xếp yêu cầu, các từ được in cách nhau một dấu cách.

Sample Input 0

```
php aba aaaa nguyen huu hoc pop
```

Sample Output 0

```
php aba pop aaaa
```

[Xâu ký tự]. Bài 13. Tần suất các từ xuất hiện trong xâu.

Cho một xâu ký tự S bao gồm các chữ cái và dấu cách, một từ được định nghĩa là các ký tự liên tiếp không chứa dấu cách, hãy đếm xem mỗi từ trong xâu xuất hiện bao nhiêu lần, đầu tiên hãy liệt kê các từ trong xâu kèm theo tần suất của mỗi từ theo thứ tự từ điển, sau đó liệt kê các từ trong xâu theo thứ tự xuất hiện.

Input Format

Dòng duy nhất chứa xâu S.

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

Đầu tiên in ra các từ trong xâu và tần suất của nó theo thứ tự từ điển. Sau đó bỏ trống 1 dòng và in ra các từ trong xâu và tần suất của nó theo thứ tự xuất hiện trong xâu.

Sample Input 0

```
bb aa bb cc aa bb cc
```

Sample Output 0

```
aa 2  
bb 3  
cc 2
```



```
bb 3
aa 2
cc 2
```

[Xâu ký tự]. Bài 14. Từ xuất hiện nhiều nhất, ít nhất

Cho một xâu ký tự S bao gồm các chữ cái và dấu cách, một từ được định nghĩa là các ký tự liên tiếp không chứa dấu cách. Hãy tìm từ có số lần xuất hiện nhiều nhất và ít nhất trong xâu, nếu có nhiều từ có cùng số lần xuất hiện nhiều nhất hoặc ít nhất thì chọn từ có thứ tự từ điển lớn nhất làm kết quả.

Input Format

Dòng duy nhất chứa xâu S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

Dòng đầu tiên in ra từ có số lần xuất hiện nhiều nhất. Dòng thứ 2 in ra từ có số lần xuất hiện ít nhất.

Sample Input 0

```
aa bb cc aa bb aa aa cc
```

Sample Output 0

```
aa 4
cc 2
```

[Xâu ký tự]. Bài 15. Kiểm tra xâu con

Cho 2 xâu S và T chỉ bao gồm các chữ cái in hoa, in thường và dấu cách. Hãy kiểm tra xem xâu T có phải là xâu con của xâu S hay không?

Input Format

Dòng đầu tiên là xâu S . Dòng thứ 2 là xâu T .

Constraints

$1 \leq \text{len}(S) \leq 100000$; $1 \leq \text{len}(T) \leq 100000$;

Output Format

In ra YES nếu xâu T xuất hiện trong xâu S, ngược lại in ra NO.

Sample Input 0

```
aabcacbdbaa  
cacb
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 16. Chuẩn hóa tên 1

Cho một xâu là tên người chỉ bao gồm các ký tự là chữ cái và dấu cách, giữa các từ trong câu có thể tồn tại nhiều dấu cách hãy chuẩn hóa tên người bằng cách viết hoa chữ cái đầu tiên của từng từ và viết thường các chữ cái còn lại. Xâu tiếp theo là ngày sinh của người này, bao gồm ngày tháng năm phân cách nhau bằng dấu '/', hãy chuẩn hóa ngày sinh của người này về đúng dạng dd/mm/yyyy.

Input Format

Dòng đầu tiên là tên người; Dòng thứ 2 là ngày sinh;

Constraints

Xâu ký tự tên người có không quá 2000 ký tự; Xâu ngày sinh có không quá 10 ký tự. Dữ liệu đảm bảo có dấu '/' giữa ngày, tháng, năm.

Output Format

Dòng đầu tiên in ra tên người sau khi chuẩn hóa. Dòng thứ 2 in ra ngày sinh sau khi đưa về dạng chuẩn dd/mm/yyyy.

Sample Input 0

```
hoang  dinh NAm  
20/5/1999
```

Sample Output 0

```
Hoang Dinh Nam  
20/05/1999
```

[Xâu ký tự]. Bài 17. Chuẩn hóa tên 2.

Cho một xâu là tên người chỉ bao gồm các kí tự là chữ cái và dấu cách, giữa các từ trong câu có thể tồn tại nhiều dấu cách hãy chuẩn hóa tên người theo 2 mẫu được yêu cầu trước. Xem output để rõ hơn về cách chuẩn hóa.

Input Format

Dòng duy nhất là tên người.

Constraints

Xâu kí tự tên người có không quá 1000 kí tự;

Output Format

Dòng đầu tiên in ra theo mẫu chuẩn hóa 1. Dòng thứ 2 in ra theo mẫu chuẩn hóa 2.

Sample Input 0

```
hoang  dINH  NaM
```

Sample Output 0

```
Hoang Dinh, NAM  
NAM, Hoang Dinh
```

[Xâu ký tự]. Bài 18. Tạo email và mật khẩu.

Hiện tại trường đại học XYZ tổ chức cấp email cho sinh viên mới nhập học. Email và mật khẩu sẽ được cấp dựa trên tên của sinh viên và ngày sinh của sinh viên đó. Bạn hãy viết chương trình để cấp tài khoản theo yêu cầu như sau, tên email được tạo bằng cách lấy tên của sinh viên và ghép với các chữ cái đầu tiên của họ và tên đệm, tất các ký tự trong email đều ở dạng in thường, ví dụ sinh viên có tên "Nguyen Van Long" sẽ được cấp email "longnv@xyz.edu.vn". Mật khẩu sẽ dựa trên ngày sinh của sinh viên đó, bằng cách ghép ngày tháng năm lại với nhau, ví dụ sinh viên sinh ngày 27/04/2002 sẽ có mật khẩu là 2742002.

Input Format

Dòng đầu tiên là số lượng sinh viên cần cấp email N. N dòng tiếp theo, mỗi dòng gồm 2 phần là họ tên và ngày sinh, ngày sinh của sinh viên đã ở dạng chuẩn dd/mm/yyyy. Giữa các từ có thể có nhiều dấu cách, từ cuối cùng trong dòng là thông tin ngày sinh của sinh viên(xâu kí tự không có dấu cách chuẩn dd/mm/yyyy).

Constraints

$1 \leq N \leq 5000$; Dòng thông tin của sinh viên không quá 1000 kí tự, dữ liệu đảm bảo thông tin cuối cùng trong dòng là ngày sinh của sinh viên.

Output Format

Đối với mỗi sinh viên in ra thông tin trên 2 dòng, dòng thứ 1 là tên email, dòng thứ 2 là mật khẩu.

Sample Input 0

```
1
nguyen VAN NAM 22/07/2002
```

Sample Output 0

```
namnv@xyz.edu.vn
2272002
```

[Xâu ký tự]. Bài 19. Tạo email và mật khẩu 2.

- Trường đại học ABC tổ chức cấp email cho sinh viên mới nhập học. Email và mật khẩu sẽ được cấp dựa trên tên của sinh viên và ngày sinh của sinh viên đó. Bạn hãy viết chương trình để cấp tài khoản theo yêu cầu như sau, tên email được tạo bằng cách lấy tên của sinh viên và ghép với các chữ cái đầu tiên của họ và tên đệm tất các ký tự trong email đều ở dạng in thường, ví dụ sinh viên có tên "Nguyen Van Long" sẽ được cấp email "longnv@xyz.edu.vn".
- Mật khẩu sẽ dựa trên ngày sinh của sinh viên đó, bằng cách ghép ngày tháng năm lại với nhau, ví dụ sinh viên sinh ngày 27/04/2002 sẽ có mật khẩu là 2742002. Ngoài ra sẽ có những trường hợp sinh viên bị trùng tên email, ví dụ, sinh viên "Nguyen Van Long" sẽ được cấp email "longnv@xyz.edu.vn", sinh viên tên "Ngo Van Long" cũng sẽ được cấp email "longnv@xyz.edu.vn", vì thế nhà trường quy định, theo thứ tự tên trong danh sách, nếu email được cấp của sinh viên hiện tại đã được cấp cho một sinh viên trước đó thì thêm số thứ tự vào tên email.

Input Format

- Dòng đầu tiên là số lượng sinh viên cần cấp email N.
- N dòng tiếp theo, mỗi dòng gồm 2 phần là họ tên và ngày sinh, ngày sinh của sinh viên đã ở dạng chuẩn dd/mm/yyyy. Giữa các từ có thể có nhiều

dấu cách, từ cuối cùng trong dòng là thông tin ngày sinh của sinh viên(xâu kí tự không có dấu cách chuẩn dd/mm/yyyy).

Constraints

- $1 \leq N \leq 5000$
- Dòng thông tin của sinh viên không quá 1000 kí tự, dữ liệu đảm bảo thông tin cuối cùng trong dòng là ngày sinh của sinh viên.

Output Format

In ra ra email và mật khẩu được cấp của mỗi sinh viên trên 2 dòng. Chú ý các sinh viên email bị trùng tên sẽ được thêm số thứ tự vào sau.

Sample Input 0

```
4
nguyen van LONG 20/10/2002
ngo Vang Long 20/12/2002
Nguyen Vu Long 22/12/2003
ho van nam 17/09/2002
```

Sample Output 0

```
longnv@xyz.edu.vn
20102002
longnv2@xyz.edu.vn
20122002
longnv3@xyz.edu.vn
22122003
namhv@xyz.edu.vn
1792002
```

[Xâu ký tự]. Bài 20. Ngôn ngữ lập trình Python.

Trong giờ học ngôn ngữ lập trình Python, thầy giáo có đưa ra một bài toán cho Tèo, bài toán như sau "Cho một xâu kí tự S, Tèo hãy xác định xem liệu rằng có thể xóa đi bất kì một kí tự nào trong xâu S ở một vị trí bất kì và xâu S trở thành từ "python" được hay không? Biết rằng Tèo có thể xóa bất kì một kí tự nào nhưng không thể hoán đổi vị trí các kí tự trong xâu S ban đầu". Hiện nay vì Tèo đang bận gói bánh chưng phụ thầy mẹ nên nhờ bạn giải giúp. Hy vọng bạn có thể giúp được Tèo !!!

Input Format

Một xâu kí tự S chỉ bao gồm các kí tự in hoa hoặc in thường.

Constraints

Xâu S có độ dài không quá 1000 kí tự.

Output Format

In ra YES nếu có thể tạo ra xâu "python" bằng cách xóa đi các kí tự trong xâu, ngược lại in NO.

Sample Input 0

```
rRMssizYyAbYPbfYPwaamZFTLSowSqCewiGIKymcnlVUuQqpXGinxSuPooRfUljsZhjn  
cyvWeDdgoOrDHeXlMbKvThvDZLnFiteaOnWapaPjmHBFlqEwkWRWTSzZnyKkpNTpidc  
XxxptIenHGKovBporbkaopJyLBZNRczxGtZPMoQmgrpiihYRdGYUSRChscdQpVZYdaTA  
qyDBfSrBmqSEDjeSysMKIuQdggY0xXuMqlEigpllWAYPOGjHlRCraPTuPgByGDokiPc  
oFHLhhVlIeGvzssrOWVetikhntuerNysZhCccZyIuPUjhBOIvYnHYopsUZuptCQoxYBS  
odMATxERREaGgNxtSjuOdyQCFlxYgGxsswbDhHWERCnjzdycpZYfCsvPPVgNqFNbLHeg  
CSBNbDzhSDjGoPKLZCQDvfKeHokfDKWQwOZKhTfTXGDCHtpopuyXabGgbyncHVklPzrq  
tLGRUZDKKvGkZaeZaVRZvfeznzbGwnNwprUEQuupkVKrzzHENzLZCITPGYsZPwgwmzc  
DCBkIVprSFwPISLGjAmZbcyIUOfmmuNDdDDDPGziCzztvi0sHGjIzysPLyTczfWrWalb  
eLgnfLVhzLPfXhoQwIjVMjBAxyVbSczfHUXaRzBVmeWvLIgpQBDgZFtgeHLHZWMIRxis  
cYqTyKNuzucMczDycPwMYNYUHCVBuZZktqxtxNWYjyTQFAcPoKteBGTxhcdezFVzuNaz  
iplGPTzZpNFVQaxEQwxdatBjRCnccjjYLTfWbVgeiraCDLGtEpgkOoWnGvawBXvtTUM  
AAnsWPmxFZqeLCQiPRzXuMhQMSUaslbNNgplnrVnpcbWcyZREbKSCArfKYblBLATNuyc  
AQEXuf
```

Sample Output 0

YES

[Xâu ký tự]. Bài 21. Tập từ chung của 2 xâu

Cho 2 xâu kí tự S và T. Hãy liệt kê các từ xuất hiện ở cả 2 xâu mà không phân biệt hoa thường, mỗi từ được liệt kê một lần theo thứ tự từ điển tăng dần.

Input Format

Dòng đầu tiên chứa xâu S. Dòng thứ hai chứa xâu T.

Constraints

$1 \leq \text{len}(S) \leq 10000$; $1 \leq \text{len}(T) \leq 10000$; Xâu S và T chỉ bao gồm chữ cái và dấu cách.

Output Format

In ra các từ xuất hiện ở cả 2 xâu theo thứ tự từ điển tăng dần, mỗi từ được liệt kê 1 lần và in ra ở dạng chữ in thường.

Sample Input 0

```
abc abc abcd abcd AB
abC CD ZAH abd ABcD
```

Sample Output 0

```
abc abcd
```

[Xâu ký tự]. Bài 22. Tập từ riêng của 2 xâu.

Cho 2 xâu ký tự S và T. Hãy liệt kê các từ xuất hiện ở xâu S và không xuất hiện ở xâu T mà không phân biệt hoa thường, mỗi từ được liệt kê một lần theo thứ tự từ điển tăng dần.

Input Format

Dòng đầu tiên chứa xâu S. Dòng thứ hai chứa xâu T.

Constraints

$1 \leq \text{len}(S) \leq 10000$; $1 \leq \text{len}(T) \leq 10000$; Xâu S và T chỉ bao gồm chữ cái và dấu cách.

Output Format

In ra các từ theo thứ tự từ điển, các từ cách nhau một dấu cách và in theo thứ tự từ điển tăng dần.

Sample Input 0

```
aa AA abc abcd zzaA
ABc ABCd uuzka aka opl
```

Sample Output 0

```
aa zzaa
```

[Xâu ký tự]. Bài 23. Ghép xâu

Cho các từ chỉ bao gồm chữ cái in hoa và in thường, bạn được yêu cầu ghép các từ này lại với nhau sao cho từ được ghép lại sau cùng có thứ tự từ điển lớn nhất.

Input Format

Dòng đầu tiên là số nguyên dương N - số lượng từ; Dòng thứ 2 gồm N từ, mỗi từ cách nhau một dấu cách.

Constraints

$1 \leq N \leq 1000$;

Output Format

In ra xâu sau khi ghép.

Sample Input 0

```
4
ab abc abcd aa
```

Sample Output 0

```
abcdabcabaa
```

[Xâu ký tự]. Bài 24. Xâu có đầu cuối giống nhau.

Cho xâu S chỉ bao gồm các ký tự in thường, hãy đếm số lượng xâu con (xâu ký tự chứa các chữ cái liên tiếp) mà có ký tự đầu và ký tự cuối giống nhau.

Input Format

Một dòng duy nhất chứa xâu S chỉ bao gồm chữ cái in thường.

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra đáp án của bài toán.

Sample Input 0

```
abcda
```

Sample Output 0

```
6
```

Explanation 0

Các xâu con có ký tự đầu cuối giống nhau : a, b, c, d, a, abcda

[Xâu ký tự]. Bài 25. Số lớn nhất, nhỏ nhất.

Cho số tự nhiên m và số nguyên s không âm. Nhiệm vụ của bạn là tìm số bé nhất và lớn nhất có m chữ số và tổng chữ số bằng s .

Input Format

Một dòng gồm 2 số m và s

Constraints

$$1 \leq m \leq 100; 0 \leq s \leq 900$$

Output Format

In ra số bé nhất, lớn nhất có thể đạt được, mỗi số in ra trên 1 dòng. Nếu không có đáp án thì in ra 1 dòng "NOT FOUND".

Sample Input 0

```
2 15
```

Sample Output 0

```
69
96
```

Sample Input 1

```
2 30
```

Sample Output 1

```
NOT FOUND
```

[Xâu ký tự]. Bài 26. Xâu con liên tiếp các kí tự giống nhau.

Cho một xâu kí tự S chỉ bao gồm các chữ cái in thường, hãy tìm xâu con liên tiếp chứa các kí tự giống nhau dài nhất, nếu có nhiều xâu con thỏa mãn thì chọn xâu con có thứ tự từ điển lớn nhất.

Input Format

Một dòng duy nhất chứa xâu S.

Constraints

$$1 \leq \text{len}(S) \leq 100000;$$

Output Format

In ra xâu con tìm được.

Sample Input 0

```
zzaabc
```

Sample Output 0

```
zz
```

[Xâu ký tự]. Bài 27. Xâu con liên tiếp các kí tự khác nhau.

Cho một xâu kí tự S chỉ bao gồm các chữ cái in thường, hãy tìm xâu con liên tiếp sao cho không có 2 kí tự nào liên kế giống nhau có độ dài lớn nhất, nếu có nhiều xâu con thỏa mãn thì chọn xâu con có thứ tự từ điển lớn nhất.

Input Format

Một dòng duy nhất chứa xâu S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra xâu con là kết quả của bài toán

Sample Input 0

```
zzaabc
```

Sample Output 0

```
abc
```

[Xâu ký tự]. Bài 28. Số lớn nhất xuất hiện trong xâu.

Cho một xâu kí tự S chỉ bao gồm các chữ cái và chữ số, hãy tìm số lớn nhất xuất hiện trong xâu. Ví dụ $S = \text{"abc123bba567ajs50kkf099"}$ thì số lớn nhất xuất hiện trong xâu là số 567, chú ý trong trường hợp một số có các số 0 đứng trước thì ta loại bỏ các số 0 ở đầu, trong ví dụ số 099 ta tính là số 99, 000 ta tính là số 0.

Input Format

Một dòng duy nhất chứa xâu S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra số lớn nhất xuất hiện trong xâu, dữ liệu đảm bảo có ít nhất 1 số xuất hiện trong xâu.

Sample Input 0

```
abc123bba567ajsjsj50kkf099
```

Sample Output 0

```
567
```

[Xâu ký tự]. Bài 29. Tính tổng các số xuất hiện trong xâu.

Cho một xâu ký tự S chỉ bao gồm các chữ cái và chữ số, hãy tìm tổng các số xuất hiện trong xâu. Ví dụ xâu ký tự "abcd123aad22a05" có tổng các số trong xâu là $123 + 22 + 5 = 150$. Trong trường hợp các số bắt đầu bằng số 0 thì ta loại bỏ các số đó, dữ liệu đảm bảo tổng các số xuất hiện trong xâu không vượt quá 10^{18} .

Input Format

Một dòng duy nhất chứa xâu S .

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra tổng các số trong xâu.

Sample Input 0

```
abcd123aad22a05
```

Sample Output 0

```
150
```

[Xâu ký tự]. Bài 30. Ghép số.

Cho một xâu ký tự S chỉ bao gồm các chữ cái và chữ số, hãy thực hiện tách các số xuất hiện trong xâu ra và ghép lại với nhau để tạo thành một số tự nhiên lớn nhất. Ví dụ với xâu $S = "123abcd234kkf11"$ thì ta sẽ tách được các số 123, 234 và 11, trong trường hợp các số bắt đầu bằng chữ số 0 thì ta loại bỏ các số 0 vô

nghĩa này. Sau đó ghép lại với nhau thành số 23412311 là số lớn nhất có thể đạt được.

Input Format

Một dòng duy nhất chứa xâu S.

Constraints

$1 \leq \text{len}(S) \leq 100000$;

Output Format

In ra số lớn nhất ghép được.

Sample Input 0

```
123abcda234kkf11
```

Sample Output 0

```
23412311
```

[Xâu ký tự]. Bài 31. Tổng chữ số của số nguyên

Cho một số nguyên không âm N, hãy tính tổng các chữ số của N.

Input Format

Số nguyên không âm N.

Constraints

N có không quá 1000 chữ số.

Output Format

In ra tổng các chữ số của N.

Sample Input 0

```
123456789
```

Sample Output 0

```
45
```

[Xâu ký tự]. Bài 32. Số đẹp 1

Một số được coi là số đẹp nếu nó có tất cả các chữ số là số nguyên tố và tổng các chữ số của nó cũng là số nguyên tố. Hãy viết chương trình kiểm tra số nguyên dương N cho trước có phải là số đẹp hay không?

Input Format

Số nguyên dương N

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N là số đẹp, ngược lại in ra NO.

Sample Input 0

```
373
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 33. Số đẹp 2

Một số được coi là số đẹp nếu nó là số thuận nghịch và chứa ít nhất 1 chữ số 6. Hãy viết chương trình kiểm tra số nguyên dương N cho trước có phải là số đẹp hay không?

Input Format

Số nguyên dương N

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N là số đẹp, ngược lại in ra NO.

Sample Input 0

```
6433987866217635596629171229463963223693649221719266955367126687893346
```

Sample Output 0

YES

[Xâu ký tự]. Bài 34. Số đẹp 3

Một số được coi là số đẹp nếu nó có các chữ số không giảm từ trái qua phải hoặc không tăng từ trái quá phải, ví dụ số 1233333345 là số không giảm, số 555555543 là số không tăng. Hãy viết chương trình kiểm tra số nguyên dương N cho trước có phải là số đẹp hay không?

Input Format

Số nguyên dương N .

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N là số đẹp ngược lại lại ra NO.

Sample Input 0

12377788888888888888888888888888888888888888888889

Sample Output 0

YES

Sample Input 1

8128777888888889

Sample Output 1

NO

[Xâu ký tự]. Bài 35. Số chia hết cho 6

Kiểm tra một số nguyên N có phải là số chia hết cho 6 hay không? Số chia hết cho 6 nếu nó chia hết cho cả 2 và 3.

Input Format

Số nguyên dương N .

Constraints

N có không quá 1000 chữ số.

Output Format

In YES nếu N là số chia hết cho 6, ngược lại in NO.

Sample Input 0

```
360
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 36. Số chia hết cho 4

Kiểm tra xem một số nguyên không âm N có chia hết cho 4 hay không, số chia hết cho 4 là số có nhiều hơn 1 chữ số mà có 2 chữ số tận cùng chia hết cho 4 hoặc là số có 1 chữ số chia hết cho 4.

Input Format

Số nguyên không âm N.

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N là số chia hết cho 4, ngược lại in ra NO.

Sample Input 0

```
516
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 37. Số chia hết cho 15

Kiểm tra xem một số nguyên không âm N có chia hết cho 15 hay không? Số chia hết cho 15 nếu nó chia hết cho cả 3 và 5.

Input Format

Số nguyên không âm N.

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N chia hết cho 15, ngược lại in ra NO.

Sample Input 0

```
150
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 38. Số chia hết cho 25

Kiểm tra xem một số nguyên không âm N có chia hết cho 25 hay không? Số chia hết cho 25 nếu nó có 2 chữ số tận cùng là 2 số 0 hoặc là số chia hết cho 25.

Input Format

Số nguyên không âm N.

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N chia hết cho 25, ngược lại in ra NO.

Sample Input 0

```
150
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 39. Số chia hết cho 11

Kiểm tra xem một số nguyên không âm N có chia hết cho 11 hay không? Để kiểm tra 1 số có chia hết cho 11 hay không bạn chỉ cần tính tổng các chữ số ở vị trí chẵn, tổng các chữ số ở vị trí lẻ và lấy độ chênh lệch của 2 tổng này chia cho 11, nếu độ chênh lệch này chia hết thì số ban đầu chia hết cho 11. Ví dụ số $N = 8580$ có tổng các số ở vị trí chẵn là 5, tổng các số ở vị trí lẻ là 16, độ chênh lệch giữa 2 tổng này là 11 vì thế số 8580 chia hết cho 11.

Input Format

Số nguyên không âm N.

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N chia hết cho 11 ngược lại in ra NO.

Sample Input 0

```
121
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 40. Số nhị phân chia hết cho 5

Cho số nhị phân N, hãy kiểm tra xem N có chia hết cho 5 hay không.

Input Format

Số N ở dạng nhị phân

Constraints

N có không quá 1000 bit.

Output Format

In ra YES nếu N chia hết cho 5, ngược lại in ra NO.

Sample Input 0

```
1010
```

Sample Output 0

```
YES
```

Sample Input 1

```
1010
```

Sample Output 1

```
YES
```

Sample Input 2

```
1010
```

Sample Output 2

```
YES
```

[Xâu ký tự]. Bài 41. Số nhị phân chia hết cho 2^K

Cho số nguyên dương N được biểu diễn được dạng số nhị phân, hãy kiểm tra xem N có chia hết cho 2^K hay không, Ví dụ $N = 11000$ (24 hệ thập phân) chia hết cho 2^3 .

Input Format

Dòng đầu tiên là số N có không quá 1000 bit. Dòng thứ 2 là số nguyên dương K .

Constraints

N có không quá 1000 bit. $1 \leq K \leq 1000$.

Output Format

In ra YES nếu N chia hết cho 2^K , ngược lại in ra NO.

Sample Input 0

```
101010010000  
3
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 42. Phép chia dư

Cho 2 số N và M , hãy tìm số dư khi chia N cho M . Để tính số dư của 2 số N và M , trong trường hợp N là 1 số nguyên lớn, ta có thể dùng kiến thức toán học sau. Ví dụ bạn có $N = 12345$ và $M = 3$, bạn có thể duyệt từng chữ số của N từ trái qua phải và duy trì số dư $r = 0$ ban đầu, khi gặp số 1, $r = r * 10 + 1$, sau đó lấy $r \% 3 = 1$, khi gặp 2, $r = r * 10 + 2 = 12$, $r \% 3 = 0$,... tương tự như vậy cho tới khi gặp số cuối cùng của N , giá trị của r khi đó chính là số dư khi chia N cho M .

Input Format

Dòng đầu tiên là số nguyên dương N. Dòng thứ 2 là số nguyên dương M.

Constraints

N có không quá 1000 chữ số; M là 1 số nguyên 64 bit.

Output Format

In ra kết quả của bài toán

Sample Input 0

```
33067946071531150754233004290758406156224088702123385775727721812560
69272812701805311820389008009780734973744548365667433775055949046328
48825152841886908750331356498961889280542914939799031248188994530520
34828440852665076293856223903153549522293752626469246456263469220701
54833962015007974895803528527845987442551014642311465145892231538215
33638674181894270625068338371026309043199729843644081432642072639241
48697330179177840468429040754651164286732641405984220989893094158917
76514234299243146384082057077238333807388939759280011878478370039646
5644597065301244994051135101466785516990398581
9999999999998156
```

Sample Output 0

```
7481318352255865
```

[Xâu ký tự]. Bài 43. Ước chung lớn nhất

Cho 2 số N và M, hãy tìm ước chung lớn nhất của N và M. Chú ý ở đây N là một số nguyên lớn và M là một số nguyên 64 bit. Gợi ý : Dùng thuật toán Euclid.

Input Format

Dòng đầu tiên là số nguyên dương N. Dòng thứ 2 là số nguyên dương M.

Constraints

N có không quá 1000 chữ số; M là 1 số nguyên 64 bit.

Output Format

In ra UCLN của N và M

Sample Input 0

```
100
20
```

Sample Output 0

[Xâu ký tự]. Bài 44. Lũy thừa với cơ số lớn

Cho 2 số N và M , hãy tính $N^M \bmod (10^9 + 7)$. Trong đó N là một số nguyên lớn, M là một số nguyên 64 bit. Gợi ý : Để làm được bài này các bạn phải sử dụng lũy thừa nhị phân để tránh Time Limit.

Input Format

Dòng đầu tiên là số nguyên dương N . Dòng thứ 2 là số nguyên dương M .

Constraints

N có không quá 1000 chữ số; M là 1 số nguyên 64 bit.

Output Format

In ra kết quả của bài toán

Sample Input 0

```
2
10
```

Sample Output 0

```
1024
```

[Xâu ký tự]. Bài 45. Chữ số cuối cùng

Cho số nguyên dương N là số nguyên lớn. Hãy tìm chữ số cuối cùng của 2008^N . Hay nói cách khác đề bài yêu cầu bạn tìm $2008^N \bmod 10$. Gợi ý, $2008^N \bmod 10 = (2008 \bmod 10)^N \bmod 10 = 8^N \bmod 10$. Mà 8 là lũy thừa của 2, vì thế chữ số cuối cùng của 8^N cũng sẽ tuần hoàn với chu kỳ 4.

Input Format

Số nguyên dương N .

Constraints

N có không quá 1000 chữ số.

Output Format

In ra đáp án của bài toán.

Sample Input 0

2

Sample Output 0

4

Explanation 0

$2008^2 = 4032064$

[Xâu ký tự]. Bài 46. Tìm số dư

Yêu cầu tính $(1^n + 2^n + 3^n + 4^n) \% 5$, trong đó n là số nguyên lớn.

Input Format

Số nguyên không âm n .

Constraints

n có không quá 1000 chữ số.

Output Format

In ra kết quả của bài toán

Sample Input 0

0

Sample Output 0

4

[Xâu ký tự]. Bài 47. Ước chung lớn nhất

Cho ba số a, x, y . Nhiệm vụ của bạn là tìm ước số chung lớn nhất của hai số P và Q , trong đó P lặp lại x lần số a và Q lặp lại y lần số a . Ví dụ $a = 2, x = 3, y = 2$ thì $P = 222, Q = 22$. Khi đó $\text{UCLN}(P, Q) = 2$

Input Format

Dòng đầu tiên là số a . Dòng số 2 là số x . Dòng số 3 là số y .

Constraints

$$1 \leq a, x, y \leq 10^{18}$$

Output Format

In ra đáp án của bài toán

Sample Input 0

```
4
10
8
```

Sample Output 0

```
44
```

[Xâu ký tự]. Bài 48. Số may mắn

Một số nguyên không âm n được gọi là số may mắn nếu tổng các chữ của n bằng 9 hoặc tổng các chữ số của n là số may mắn. Ví dụ các số 9, 108, 279 là các số may mắn, còn các số 19, 289 không phải là số may mắn.

Input Format

Số nguyên dương N .

Constraints

N có không quá 1000 chữ số.

Output Format

In ra YES nếu N là số may mắn, ngược lại in ra NO.

Sample Input 0

```
18
```

Sample Output 0

```
YES
```

[Xâu ký tự]. Bài 49. Xóa cụm 111

Theo quan niệm của người Việt, số 7 là một số không may mắn, vì thế bạn được yêu cầu xóa cụm 111 trong xâu nhị phân ban đầu, cụm 111 chính là biểu diễn ở dạng nhị phân của số 7.

Input Format

Xâu nhị phân chỉ bao gồm số 0 và 1.

Constraints

Xâu có độ dài không quá 1000.

Output Format

In ra xâu sau khi xóa cụm 111, chú ý khi bạn xóa 1 cụm 111 khỏi xâu ban đầu, các kí tự ở bên trái và bên phải cụm 111 này lại trở thành các kí tự liền kề nhau. Nếu sau khi xóa xâu trở thành rỗng thì in ra "EMPTY";

Sample Input 0

```
1000111
```

Sample Output 0

```
1000
```

Sample Input 1

```
111
```

Sample Output 1

```
EMPTY
```

[Xâu ký tự]. Bài 50. Tích giai thừa các chữ số.

Axe chơi một trò chơi với Lina. Họ định nghĩa hàm $F(x)$ với số x nguyên dương là tích giai thừa các chữ số của x . Ví dụ $F(135) = 1! * 3! * 5! = 720$. Đầu tiên, họ chọn một số a có n chữ số và có ít nhất một chữ số lớn hơn 1, có thể có chữ số không ở đầu. Sau đó họ tìm một số nguyên dương x lớn nhất thỏa mãn: 1. x không chứa chữ số 0 hoặc 1 2. $F(x) = F(a)$ Hãy giúp Axe và Lina tìm ra được số đó.

Input Format

Dòng duy nhất chứa số a có n chữ số.

Constraints

$1 \leq n \leq 1000$.

Output Format

In ra số X tìm được

Sample Input 0

```
1234
```

Sample Output 0

```
33222
```

[Xâu ký tự]. Bài 51. Nén xâu

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho xâu ký tự S , nhiệm vụ của bạn là nén các ký tự liên kế của S lại bằng gom các ký tự liên kế giống nhau và ghi số lần xuất hiện của ký tự đó vào sau. Ví dụ $S = aaabbbccccdzaa \Rightarrow a3b3c4d1z1a2$.

Input Format

Dòng duy nhất chứa xâu S

Constraints

- $1 \leq \text{len}(S) \leq 10^6$
- S chỉ bao gồm các ký tự in thường

Output Format

In ra xâu được nén

Sample Input 0

```
aacducuucdcabczubduuubaaa
```

Sample Output 0

```
a2c1d1u1c1u2c1d1c1a1b1c1z1u1b1d1u3b1a3
```

[Xâu ký tự]. Bài 52. Giải nén xâu

- Problem
- Submissions
- Discussions

Cho xâu S bao gồm các ký tự và số lần xuất hiện tương ứng của nó, bạn hãy giải nén xâu. Ví dụ xâu S = a2b3z2u1 sẽ được giải nén thành aabbzbzzu, S = z11 sẽ được giải nén thành zzzzzzzzzzz. Dữ liệu đảm bảo theo sau 1 kí tự sẽ luôn là 1 số nguyên dương int 32 bit.

Input Format

Dòng duy nhất chứa xâu S

Constraints

- $2 \leq \text{len}(S) \leq 10^5$
- S chỉ bao gồm kí tự in thường và chữ số

Output Format

In ra sâu được giải nén

Sample Input 0

h7j4w6z1z8z10z8z1u8u6u1

Sample Output 0

hhhhhhhjjjwwwwwwwzzzzzzzzzzzzzzzzzzzzzzzzzzzzzzuuuuuuuuuuuuuuuuuu

[Xâu ký tự]. Bài 53. Score

- **Problem**
- **Submissions**
- **Discussions**

Cho danh sách các trận đấu, nhiệm vụ của bạn là cập nhật xem mỗi đội bóng ghi được tất cả bao nhiêu bàn thắng và liệt kê tên đội bóng kèm số bàn thắng mà đội này ghi được theo thứ tự giảm dần về số bàn thắng, nếu 2 đội có cùng số lượng bàn thắng thì in tên đội bóng theo từ điển tăng dần . Thông tin các trận đấu được cho dưới dạng $X - b Y$ trong đó X, Y là tên của 2 đội bóng, a là số bàn thắng của đội X và b là số bàn thắng của đội Y . Ví dụ Chelsea 3 - 4 Arsenal.

Input Format

Gồm nhiều dòng, mỗi dòng là thông tin 1 trận đấu.

Constraints

Có không quá 100000 lượt đấu

Output Format

In ra kết quả của bài toán

Sample Input 0

```
Barca 3 - 18 Man City
28Tech Football Club 5 - 19 Newcastle United
Chelsea 6 - 0 Newcastle United
28Tech Football Club 18 - 12 Liverpool
Manchester United 13 - 13 Aston Villa
Arsenal 19 - 0 Aston Villa
Barca 19 - 15 Man City
Manchester United 19 - 6 Aston Villa
Manchester United 19 - 13 Aston Villa
Chelsea 15 - 3 Liverpool
```

Sample Output 0

```
Manchester United 51
Man City 33
Aston Villa 32
28Tech Football Club 23
Barca 22
Chelsea 21
Arsenal 19
Newcastle United 19
Liverpool 15
```

[Xâu ký tự]. Bài 54. Anagram

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho 2 xâu S và T, 2 xâu này được gọi là anagram nếu chúng có số lần xuất hiện của các ký tự trong cả 2 xâu giống nhau, thứ tự xuất hiện của các ký tự có thể khác nhau. Nhiệm vụ của bạn là kiểm tra xem 2 xâu S và T có phải là anagram hay không?

Input Format

- Dòng 1 là số bộ test T
- T dòng tiếp theo mỗi dòng chứa 2 xâu S và T

Constraints

- $1 \leq T \leq 1000$
- $1 \leq \text{len}(S) \leq \text{len}(T) \leq 10^5$
- S và T chỉ bao gồm các ký tự in thường

Output Format

Đối với mỗi test case in ra YES nếu S và T là anagram, ngược lại in ra NO

Sample Input 0

```
10
acduzz dazzc
zdbbz dbbzc
zbcaudcbu duaucdczc
zdzadc zzdcda
czbzzau zzdzdau
zadbdbc dbbdbc
accubczc zcubcccca
zbacuc zcuaz
zbbbd zbcub
daaad addaa
```

Sample Output 0

```
YES
NO
NO
YES
NO
NO
YES
NO
NO
YES
```

[Xâu ký tự]. Bài 55. Xâu đối xứng

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho chuỗi ký tự S, bạn hãy kiểm tra xem mình có thể sắp xếp lại các ký tự trong chuỗi sao cho chuỗi S trở thành chuỗi đối xứng hay không? Ví dụ S = "zaabcczb" có thể sắp xếp lại thành "zabccbaz" và trở thành chuỗi đối xứng.

Input Format

- Dòng 1 là số bộ test T
- T dòng tiếp theo mỗi dòng là chuỗi S

Constraints

- $1 \leq T \leq 1000$
- Chuỗi S chỉ bao gồm ký tự in thường và có độ dài không quá 10^6

Output Format

Đối với mỗi test case in ra YES hoặc NO

Sample Input 0

```
10
zzcdbaca
bzzab
bbddadz
zzcabzc
cdzzda
dccdzzdd
bdzcda
aabbbaa
aadada
cbacz
```

Sample Output 0

```
NO
YES
NO
NO
NO
YES
NO
YES
YES
NO
```

[Chuỗi ký tự]. Bài 56. Số La Mã

- [Problem](#)

- [Submissions](#)
- [Discussions](#)

Số La Mã được biểu diễn thông qua 7 ký tự :

- I : 1
- V : 5
- X : 10
- L : 50
- C : 100
- D : 500
- M : 1000

Các ký tự I, X, C, M lặp lại không quá 3 lần liên tiếp, các ký tự V, D, L lặp lại không quá 1 lần liên tiếp. Khi đó ta có 6 tổ hợp đặc biệt có thể xuất hiện khi kết hợp 2 ký tự lại gồm : IV = 4, IX = 9, XL = 40, XC = 90, CD = 400, CM = 900

Bạn hãy thực hiện chuyển đổi từ số La Mã sang số thập phân

Input Format

- Dòng đầu là số bộ test T
- T dòng tiếp theo mỗi dòng là một số La Mã

Constraints

- $1 \leq T \leq 100$
- Số La Mã là một chuỗi có không quá 20 ký tự

Output Format

In ra giá trị của số La Mã tương ứng với mỗi test case

Sample Input 0

```
5
XV
L
CCLXXV
MCCLXXV
CCCXXV
```

Sample Output 0

```
15
50
275
```

1275

325

Sample Input 1

1

MMCMXCIX

Sample Output 1

3999

[Xâu ký tự]. Bài 57. 28Tech & chemistry

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

28tech là một người yêu thích hóa học, trong môn hóa học anh ta thường phải tính phân tử khối của một chất hóa học. Chất hóa học này bao gồm nhiều nguyên tử cùng với số lần xuất hiện của nó. Bạn hãy giúp 28tech tìm ra nguyên tử khối của các chất hóa học anh ta đang có.

Input Format

- 10 dòng đầu tiên của input là 10 nguyên tố hóa học xuất hiện trong các công thức hóa học kèm nguyên tử khối của nó.
- Dòng số 11 là T : số lượng công thức hóa học
- T dòng tiếp theo là công thức hóa học của T chất

Constraints

- $1 \leq T \leq 100$
- Công thức hóa học có độ dài không quá 10000, số lần xuất hiện của các nguyên tử trong phân tử không quá 10000.

Output Format

In ra phân tử khối của từng chất hóa học trên từng dòng

Sample Input 0

Al 27

Ca 40

F 19

H 1

He 4

K 39

N 14
O 16
P 31
S 32
6
H7Al6O7
Ca3Al4P7
N3He7K4
F5F6He6
He5Ca4N3
Ca6H6N7

Sample Output 0

281
445
226
233
222
344