

STRING

Bài 0. Tách từ với stringstream	2
Bài 1. Xóa từ trong câu	2
Bài 2. Tạo email	3
Bài 3. Chuẩn hóa tên 1	3
Bài 4. Chuẩn hóa tên 2	4
Bài 5.1. Xâu pangram	4
Bài 5.2. Xâu pangram 2	5
Bài 6. Đếm số từ trong xâu	5
Bài 7. Đầu cuối giống nhau	6
Bài 8. Ký tự không lặp	6
Bài 9. Lớn nhất, nhỏ nhất	7
Bài 10. Đếm số lượng câu khác nhau	7
Bài 11. Số đẹp	8
Bài 12. Đổi chỗ chữ số	8
Bài 13. Sắp đặt xâu kí tự	9
Bài 14. Tổng lớn nhất và nhỏ nhất	9
Bài 15. Tính tổng các số trong xâu	10
Bài 16. Số lớn nhất trong xâu	11
Bài 17. Đếm số xâu con	11
Bài 18. Hiệu của 2 tập từ	12
Bài 19. Biến số đẹp	12
Bài 20. Xâu con nhỏ nhất 1	14
Bài 21. Xâu con nhỏ nhất 2	15
Bài 22. Sumstring	16
Bài 23. Bảng màu RGB	16
Bài 24. Đảo từ	17
Bài 25. Tính giai thừa các chữ số	17

Bài 26. Giải mã tăng giảm	18
Bài 27. Loại bỏ 100	19
Bài 28. Số lặp lại	20
Bài 29. Số đối xứng lớn nhất	20

andre
w2804

Mọi thắc mắc và góp ý về đề bài các bạn liên hệ với mình qua địa chỉ email:

andrew168545824@gmail.com hoặc Zalo/Tele : 0965303260

Các bạn có thể tham khảo video lời giải của mình tại

<https://cutt.ly/WmI0f6O>

Bài 0. Tách từ với stringstream

Đếm số lượng từ trong xâu

Input

Dòng đầu tiên chứa số lượng test case.

Mỗi test case gồm một xâu trên 1 dòng không quá 10000 kí tự

Output

In số lượng từ của xâu trên 1 dòng

Ví dụ

Input	Output

Bài 1. Xóa từ trong câu

Thực hiện loại bỏ các từ trong 1 xâu

Input

Dòng 1 chứa xâu có không quá 1000 kí tự

Dòng 2 chứa từ cần loại bỏ có không quá 10 kí tự

Output

Thực hiện loại bỏ từ trong xâu

Ví dụ

Input	Output
lap trình python java python c lap trình trình	lap python java python c lap

Source code : <https://ideone.com/hZW5Vu>

Bài 2. Tạo email

Thực hiện tạo email từ tên người dùng

Input

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

T dòng tiếp theo mỗi dòng chứa xâu có không quá 1000 kí tự

Output

In tên email được cấp theo mẫu

Ví dụ

Input	Output
2 NguYEN VAN maNH nGUYEN thuY LinH	<u>manhnv@gmail.com</u> <u>linhnt@gmail.com</u>

Source code : <https://ideone.com/Y1cDF4>

Bài 3. Chuẩn hóa tên 1

Thực hiện viết hoa chữ cái đầu của từng từ trong tên người. Tên người là một xâu có thể không chuẩn.

Input

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

T dòng tiếp theo mỗi dòng chứa xâu có không quá 1000 kí tự

Output

In tên người đã được chuẩn hóa trên từng dòng

Ví dụ

Input	Output
2	

NguyEN VAN maNH nGUYEN thuY LinH	Nguyen Van, MANH Nguyen Thuy, LINH
--	---------------------------------------

Source code : <https://ideone.com/w019ch>

Bài 4. Chuẩn hóa tên 2

Thực hiện chuẩn hóa tên người theo mẫu. Tên người là một xâu có thể không chuẩn.

Input

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

T dòng tiếp theo mỗi dòng chứa xâu có không quá 1000 kí tự

Output

In tên người đã được chuẩn hóa trên từng dòng

Ví dụ

Input	Output
2 NguYEN VAN maNH nGUYEN thuY LinH	Manh, Nguyen Van Linh, Nguyen Thuy

Source code : <https://ideone.com/RIRogB>

Bài 5.1. Xâu pangram

Một xâu chứa đầy đủ các kí tự in thường từ a-z được gọi là xâu Pangram

Kiểm tra xâu nhập vào có phải là xâu pangram hay không

Input

Dòng duy nhất là xâu gồm các kí tự in thường không quá 1000 kí tự

Output

In YES nếu xâu là xâu pangram, ngược lại in NO

Ví dụ

Input	Output
thequickbrownfoxjumpsoverthelazydog	YES
abcdefghijklmnpzzutvlt	NO

Source code : <https://ideone.com/rHwrR6>

Bài 5.2. Xâu pangram 2

Một xâu ký tự được gọi là Pangram nếu nó chứa đầy đủ các ký tự từ 'a',..'z'. Cho xâu ký tự S và số K. Hãy xác định xem có thể thực hiện nhiều nhất K phép biến đổi các ký tự để S trở thành Pangram hay không? Mỗi phép biến đổi là một phép thay thế ký tự này bằng một ký tự khác.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào xâu ký tự S; dòng tiếp theo đưa vào số K.
- T, S, K thỏa mãn ràng buộc: $1 \leq T \leq 100$; $0 \leq K \leq 62$; $1 \leq \text{Length}(S) \leq 10^6$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	0
qwqqwqeqqwdsdadsdasadsfsdsdsdasasas	
4	1
qwqqwqeqqwdsdadsdasadsfsdsdsdasasas	
24	

Source code : <https://ideone.com/btmcBn>

Bài 6. Đếm số từ trong xâu

Input

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

T dòng tiếp theo mỗi dòng chứa xâu có không quá 1000 ký tự

Output

In ra số lượng từ trong 1 xâu trên từng dòng

Ví dụ

Input	Output
2	
Java php python C++	4
Quick fox	2

Source code : <https://ideone.com/TIM9Lq>

Bài 7. Đầu cuối giống nhau

Hãy đếm tất cả các xâu con của S có ký tự đầu và ký tự cuối giống nhau.

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

T dòng tiếp theo mỗi dòng chứa xâu có không quá 1000 ký tự

Output

In ra kết quả từng test case

Ví dụ

Input	Output
2	
abcda	5
aaa	6

Source code : <https://ideone.com/j4wW3a>

Bài 8. Ký tự không lặp

Cho xâu ký tự S. Nhiệm vụ của bạn là in ra tất cả các ký tự không lặp khác nhau trong S. Ví dụ S = "ABCDEABC" ta nhận được kết quả là "DE".

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ký tự S được viết trên một dòng.

T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{Length}(S) \leq 10^5$.

Output:

Đưa ra kết quả mỗi test theo từng dòng. In ra theo thứ tự xuất hiện.

Ví dụ

Input	Output
-------	--------

2 ABCDEAABC ABC	DE ABC
-----------------------	---------------

Source code : <https://ideone.com/5h8Wx7>

Bài 9. 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

Dòng đầu gồm 2 số m và s ($1 \leq m \leq 100, 0 \leq s \leq 900$).

Output

In ra kết quả của bài toán.

Số đầu tiên là số bé nhất, số thứ hai là số lớn nhất. Nếu không có đáp án in ra "-1 -1".

Ví dụ

Input	Output
2 15	69 96

Source code : <https://ideone.com/dkSZRF>

Bài 10. Đếm số lượng câu khác nhau

Dòng đầu tiên là số lượng câu n ($1 \leq n \leq 10^4$).

n dòng tiếp theo mỗi dòng chứa xâu có không quá 1000 kí tự

Output

In ra số lượng câu khác nhau

Ví dụ

Input	Output
5 Lap trinh DSA DSA Lap trinh Lap trinh huong doi tuong DSA DSA	3

Source code : <https://ideone.com/mocAYU>

Bài 11. Số đẹp

Một số được coi là đẹp nếu đó là số thuận nghịch và chỉ toàn các chữ số chẵn. Viết chương trình đọc vào các số nguyên dương có không quá 500 chữ số và kiểm tra xem số đó có đẹp hay không.

Input:

Dòng đầu tiên ghi số bộ test.

Mỗi bộ test viết trên một dòng số nguyên dương n không quá 500 chữ số.

Output:

Mỗi bộ test viết ra trên một dòng chữ YES nếu đó là số đẹp, chữ NO nếu ngược lại

Ví dụ

Input	Output
4	NO
123456787654321	YES
86442824468	YES
8006000444422220000222244440006008	NO
235365789787654324567856578654356786556	

Source code : <https://ideone.com/qDAiPl>

Bài 12. Đổi chỗ chữ số

Cho số tự nhiên N. Bạn chỉ được phép sử dụng nhiều nhất một phép đổi chỗ giữa 2 chữ số để nhận được số lớn nhất nhỏ hơn N. Ví dụ với số N=12435, sử dụng một phép đổi chỗ ta nhận được số lớn nhất nhỏ hơn N là 12345. Mặc dù $12354 > 12345$ nhưng ta không thể tạo ra số 12345 với chỉ một phép hoán vị. Với số N=12345 ta không có phép đổi chỗ.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- T dòng tiếp đưa các bộ test. Mỗi bộ test được viết trên một dòng là một xâu ký tự số không có ký tự '0' đầu tiên.

- Các số T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(N) \leq 10^5$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng, trong đó -1 được xem là test không có phép đổi chỗ.

Input	Output
2	12345
12435	-1
12345	

Source code : <https://ideone.com/HgpHwK>

Bài 13. Sắp đặt xâu kí tự

Cho xâu ký tự S bao gồm các ký tự 'a', ..., 'z'. Các ký tự trong S có thể lặp lại. Nhiệm vụ của bạn sắp đặt lại các ký tự trong S sao cho các ký tự kề nhau đều khác nhau.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ký tự S.
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $0 \leq \text{Length}(S) \leq 10^3$.

Output:

- Đưa ra 1 hoặc 0 nếu có thể hoặc không thể sắp đặt lại các ký tự trong xâu S thỏa mãn yêu cầu bài toán.

Input:	Output:
3	1
geeksforgeeks	1
bbbabaaacd	0
bbbbbb	

Source code : <https://ideone.com/CXT0LD>

Bài 14. Tổng lớn nhất và nhỏ nhất

Cho hai số nguyên dương $X1, X2$. Ta chỉ được phép thay đổi chữ số 5 thành 6 và ngược lại chữ số 6 thành chữ số 5 của các số $X1$ và $X2$. Hãy đưa ra tổng nhỏ nhất và tổng lớn nhất các số $X1$ và $X2$ được tạo ra theo nguyên tắc kể trên.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là cặp các số $X1, X2$.
- $T, X1, X2$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 0 \leq X1, X2 \leq 10^{18}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1100 1312
645 666	10010 11132
5466 4555	

Source code : <https://ideone.com/EQkZIx>

Bài 15. Tính tổng các số trong xâu

Cho xâu ký tự S bao gồm các ký tự 'a', ..., 'z' và các chữ số. Nhiệm vụ của bạn là hãy tính tổng các số có mặt trong xâu.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ký tự S .
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100; 0 \leq \text{Length}(S) \leq 10^5$.
- Input đảm bảo đáp án không vượt quá 10^9 .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
--------	---------

4	24
1abc23	4
geeks4geeks	100
1abc2x30yz67	123
123abc	

Source code : <https://ideone.com/nUY1k8>

Bài 16. Số lớn nhất trong chuỗi

Cho chuỗi ký tự S bao gồm các ký tự 'a', ..., 'z' và các chữ số. Nhiệm vụ của bạn là hãy tìm số lớn nhất có mặt trong chuỗi.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một chuỗi ký tự S.
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $0 \leq \text{Length}(S) \leq 10^5$.
- Input đảm bảo đáp số không vượt quá 10^9 .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
3	564
100klh564abc365bg	9
abvhd9sdnkjdfs	0
abchsd0sdhs	

Source code : <https://ideone.com/DyYHcV>

Bài 17. Đếm số chuỗi con

Cho chuỗi ký tự S và số k. Nhiệm vụ của bạn là đếm số chuỗi con của S có đúng k ký tự khác nhau. Các chuỗi con không nhất thiết phải khác nhau. Ví dụ với chuỗi S="abc" và k = 2, ta có kết quả là 2 bao gồm các chuỗi con: "ab", "bc".

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một chuỗi ký tự S và số k được viết trên một dòng.

- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq k \leq 26$; $1 \leq \text{Length}(S) \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	2
abc 2	3
aba 2	

Source code : <https://ideone.com/9VQn5X>

Bài 18. Hiệu của 2 tập từ

Cho hai chuỗi ký tự S1 và S2. Hãy viết chương trình tìm các từ chỉ xuất hiện trong S1 mà không xuất hiện trong S2. Chú ý: mỗi từ chỉ liệt kê 1 lần.

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test gồm 2 dòng, mỗi dòng ghi một chuỗi ký tự độ dài không quá 200, chỉ bao gồm các ký tự viết thường và các khoảng trống.

Kết quả: Với mỗi bộ test ghi ra các từ có trong S1 mà không có trong S2. Các từ được ghi theo thứ tự từ điển.

Ví dụ:

Input	Output
2	abcd
abc ab ab ab abcd	aaa ab abc zzz
ab abc	
aaa xyz ab zzz abc dd dd abc	
xyz dd ttt sas cdc	

Source code : <https://ideone.com/tjUllv>

Bài 19. Biển số đẹp

Biển số xe máy được quy định gồm các thành phần:

- Hai chữ số đầu là mã quản lý theo tỉnh – thành phố (ví dụ mã của Hà Nội là 29 đến 31)

- Tiếp theo là dấu gạch ngang, sau đó là một chữ cái in hoa (từ A đến Z) và một chữ số. Cặp chữ cái và chữ số này được cấp theo khu vực quận – huyện.
- Tiếp theo là một dãy 5 số gồm 2 cụm: 3 chữ số đầu và hai chữ số sau, phân tách bởi dấu chấm.

Thông thường, người ta chỉ quan tâm đến 5 chữ số cuối. Giả sử các trường hợp sau được coi là đẹp:

- Cả 5 chữ số được sắp theo thứ tự tăng chặt từ trái qua phải.
- Cả 5 chữ số đều bằng nhau
- Ba chữ số đầu bằng nhau và hai chữ số cuối bằng nhau
- Cả 5 chữ số đều là 6 và/hoặc 8 (số lộc phát).

Theo quy tắc trên, các biển số sau được coi là đẹp:

- 29-T1 123.79
- 29-T1 555.55
- 29-T1 222.33
- 29-T1 686.88

Và các biển số sau không đẹp:

- 29-T1 123.33
- 29-T1 555.54
- 29-T1 606.88

Viết chương trình kiểm tra xem các biển số xe có đẹp hay không.

Dữ liệu vào

- Dòng đầu ghi số bộ test, không quá 50
- Mỗi bộ test là một biển số. Dữ liệu vào đảm bảo biển số được viết đúng quy định.

Kết quả

- In ra kết quả kiểm tra với từng bộ test

Ví dụ

Input	Output
-------	--------

7	YES
29T1-123.45	YES
29T1-555.55	YES
29T1-222.33	YES
29T1-686.88	NO
29T1-123.33	NO
29T1-555.54	NO
29T1-606.88	

Source code : <https://ideone.com/IKZxM4>

Bài 20. Số đầy đủ

Cho một số nguyên dương lớn có nhiều hơn 20 chữ số nhưng không quá 1000 chữ số. Hãy kiểm tra xem số đó có đầy đủ tất cả các chữ số từ 0 đến 9 hay không.

Dữ liệu vào

- Dòng đầu ghi số bộ test, không quá 10
- Mỗi bộ test là một dãy ký tự có độ dài không quá 1000, không có khoảng trống

Kết quả

- Nếu dữ liệu vào không phải là một số nguyên hợp lệ (có ký tự không phải số hoặc bắt đầu bằng chữ số 0) thì in ra INVALID
- Nếu dữ liệu vào thỏa mãn đầy đủ thì in ra YES, nếu không in ra NO

Ví dụ

Input	Output
3	INVALID
01234aa32432432432534545b987978	YES
12344444444444445666666666667890	NO

324562783924723543243243242354354354333234324

Source code : <https://ideone.com/qpyVvU>

Bài 20. Xâu con nhỏ nhất 1

Cho hai xâu ký tự S1 và S2. Nhiệm vụ của bạn là hãy tìm xâu con nhỏ nhất của S1 chứa đầy đủ các ký tự của S2. Nếu không tồn tại xâu con thỏa mãn yêu cầu bài toán, hãy đưa ra -1.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là bộ đôi S1 và xâu ký tự S2 được viết trên các dòng khác nhau.
- T, S1, S2 thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{Length}(S1), \text{Length}(S2) \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	toprac
timetopractice	apzo
toc	
zoomlazapzo	
oza	

Source code : <https://ideone.com/rKOw4h>

Bài 21. Xâu con nhỏ nhất 2

Cho xâu ký tự S. Nhiệm vụ của bạn là hãy tìm độ dài xâu con nhỏ nhất của S chứa đầy đủ các ký tự của S mỗi ký tự ít nhất một lần. Ví dụ với xâu S="aabcdbcdbca" ta có xâu con nhỏ nhất là "dbca".

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ký tự S được viết trên một dòng.
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{Length}(S) \leq 10^5$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	4
aabcbcdabca	2
aaab	

Source code : <https://ideone.com/PiaVu1>

Bài 22. Sumstring

Cho chuỗi ký tự S bao gồm các chữ số. Chuỗi S được gọi là Sum String nếu tồn tại một số tự nhiên $k > 2$ sao cho ta có thể chia chuỗi S thành k chuỗi con khác nhau $S = (S_1, S_2, \dots, S_k)$ sao cho các số được tạo bởi các chuỗi con thỏa mãn điều kiện $S_i = S_{i-1} + S_{i-2}$ ($i = 3, 4, \dots, k$). Ví dụ chuỗi $S = "123415538"$ là một Sum String vì tồn tại số $k = 3$ để phân tích chuỗi S thành 3 chuỗi con $S = ("123", "415", "538")$ thỏa mãn $123 + 414 = 538$. Tương tự như vậy chuỗi $S = "12345"$ không phải là một Sum String.

Hãy kiểm tra xem S có phải là chuỗi Sum String hay không?

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các test. Mỗi test là một chuỗi ký tự số S.
- T và S thỏa mãn ràng buộc $1 \leq T \leq 100$, $3 \leq \text{length}(S) \leq 105$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
3	Yes
123415538	No
12345	Yes
1122335588143	

Source code : <https://ideone.com/15Zflw>

Bài 23. Bảng màu RGB

Ta cần tạo một chuỗi ký tự S có độ dài n . Trong đó, mỗi ký tự trong S chỉ là các ký tự R, B, hoặc G. Chuỗi ký tự nhận được có ít nhất r ký tự R, b ký tự B, g ký tự G ($r + b + g \leq n$). Hãy đếm số các chuỗi ký tự thỏa mãn yêu cầu kể trên. Ví dụ với $n=4, r=1, b=1, g=1$ ta có thể có 36 chuỗi ký tự khác nhau.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là bộ bốn số phân biệt n, r, b, g được viết trên một dòng.
- T, S, n, r, b, g thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq n \leq 20; 1 \leq r, b, g \leq N$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	36
4 1 1 1	22
4 2 0 1	

Source code : <https://ideone.com/OeIBlR>

Bài 24. Đảo từ

Cho chuỗi ký tự S . Nhiệm vụ của bạn là đảo ngược các từ trong S . Ví dụ $S = \text{"I like this program very much"}$, ta nhận được kết quả là $\text{"much very program this like I"}$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một chuỗi ký tự S .
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq \text{Length}(S) \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
--------	---------

2 I like this program very much much very program this like I	much very program this like I I like this program very much
---	--

Source code : <https://ideone.com/i3MuEA>

Bài 25. Tính 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

Dòng đầu tiên chứa số bộ test T ($T < 100$).

Mỗi test gồm một dòng chứa số n và số a ($1 \leq n \leq 15$).

Output

In ra kết quả mỗi test trên một dòng là số lớn nhất tìm được.

Input	Output
1 4 1234	33222

Source code : <https://ideone.com/ypvXG0>

Bài 26. Giải mã tăng giảm

Cho mảng $A[]$ chỉ bao gồm các ký tự I hoặc D. Ký tự I được hiểu là tăng (Increasing) ký tự D được hiểu là giảm (Decreasing). Sử dụng các số từ 1 đến 9, hãy đưa ra số nhỏ nhất được

đoán nhận từ mảng A[]. Chú ý, các số không được phép lặp lại. Dưới đây là một số ví dụ mẫu:

- A[] = "I" : số tăng nhỏ nhất là 12.
- A[] = "D" : số giảm nhỏ nhất là 21
- A[] = "DD" : số giảm nhỏ nhất là 321
- A[] = "DDIDDIID": số thỏa mãn 321654798

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ID
- T, Length(A) thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{Length}(A) \leq 9$; .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
4	12
I	21
D	321
DD	321654798
DDIDDIID	

Source code : <https://ideone.com/In2RFt>

Bài 27. Loại bỏ 100

Cho xâu ký tự S chỉ bao gồm các ký tự '0' và '1'. Nhiệm vụ của bạn là loại bỏ các xâu con "100" trong S và đưa ra độ dài lớn nhất xâu con bị loại bỏ. Ví dụ S = "1011110000" ta nhận được kết quả là 6 vì ta cần loại bỏ xâu "110000" có độ dài 6.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu ký tự nhị phân S được viết trên một dòng.
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{Length}(S) \leq 10^5$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	3
010010	6
1011110000	

Source code : <https://ideone.com/2qouhy>

Bài 28. Số lặp lại

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 $P = 222, Q = 22$.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- T dòng tiếp đưa các bộ test. Mỗi bộ test được viết trên một dòng là bộ ba số a, x, y phân biệt nhau bởi một vài khoảng trống.
- Các số T, a, x, y thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq a, x, y \leq 10^{18}$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input	Output
2	2
2 2 3	123
123 5 2	

Source code : <https://ideone.com/PkOmVR>

Bài 29. Số đối xứng lớn nhất

Cho số nguyên dương gồm N chữ số. Bạn chỉ được phép thực hiện hai thao tác:

Thao tác A: loại bỏ tất cả các chữ số giống nhau.

Thao tác B: sắp đặt lại vị trí các chữ số.

Hãy tìm số nguyên đối xứng lớn nhất có thể được tạo ra bằng cách thực hiện hai thao tác A hoặc B ở trên. Ví dụ với số $N = 1122233300000998$, ta có thể tạo ra số đối xứng lớn nhất là 910000019 bằng cách thực hiện các thao tác A, B như sau:

Thao tác A: loại bỏ các chữ số 2 ta nhận được số 1133300000998.

Thao tác A: loại bỏ các chữ số 3 ta nhận được số 1100000998.

Thao tác A: loại bỏ các chữ số 8 ta nhận được số 110000099.

Thao tác B: sắp đặt lại các số còn lại 110000099 để được số 910000019.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số nguyên dương có N chữ số.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 1000$.

Output:

- Đưa ra số nguyên lớn nhất được tạo ra bởi thao tác A, B của mỗi test theo từng dòng.

Input:	Output:
6	5
12345000	910000019
11233300000998	333
3335	63336
333566	555
33355500	

Source code : <https://ideone.com/Yu86v0>

Bài 30. Số nhị phân chia hết cho 2^x

Kiểm tra một số nguyên dương được biểu diễn dưới dạng số nhị phân có chia hết cho 2^x hay không.

Input

Dòng đầu tiên là số lượng bộ test, mỗi bộ test gồm một dòng bao gồm số nhị phân n ($1 \leq n \leq 10^{1000}$). và số tự nhiên x ($1 \leq x \leq 100$).

Output

In YES nếu n chia hết cho 2^k , ngược lại in NO.

Input	Output
2	YES
1010100101010101001010101010100000000 8	NO
1111111100001011111100000000000000000111111110000000000	
100	

Source code : <https://ideone.com/rItFVN>