

ĐỀ BÀI LUYỆN TẬP LẬP TRÌNH JAVA CƠ BẢN

BÀI A: ƯỚC SỐ CHUNG LỚN NHẤT VÀ BỘI SỐ CHUNG NHỎ NHẤT

Viết chương trình tìm ước số chung lớn nhất và bội số chung nhỏ nhất của hai số nguyên dương a,b.

Dữ liệu vào: Dòng đầu ghi số bộ test. Mỗi bộ test ghi trên một dòng 2 số nguyên a và b không quá 9 chữ số.

Kết quả: Mỗi bộ test ghi trên 1 dòng, lần lượt là USCLN, sau đó đến BSCNN.

Ví dụ:

Input	Output
2	2 204
12 34	2 3503326
1234 5678	

BÀI B: BẮT ĐẦU VÀ KẾT THÚC

Viết chương trình kiểm tra một số nguyên dương bất kỳ (2 chữ số trở lên, không quá 9 chữ số) có chữ số bắt đầu và kết thúc bằng nhau hay không. **Dữ liệu vào:** Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng số nguyên dương tương ứng cần kiểm tra. **Kết quả:** Mỗi bộ test viết ra YES hoặc NO, tương ứng với bộ dữ liệu vào

Ví dụ:

Input	Output
2	YES
12451	NO
1000012	

BÀI C: TỔNG CHỮ SỐ CHIA HẾT CHO 10

Viết chương trình kiểm tra một số có thỏa mãn tính chất tổng chữ số của nó chia hết cho 10 hay không. **Dữ liệu vào:** Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng một số nguyên dương, ít nhất 2 chữ số nhưng không quá 9 chữ số.

Kết quả: Mỗi bộ test viết ra YES hoặc NO tùy thuộc kết quả kiểm tra.

Ví dụ

Input	Output
3	NO
3333	YES
555555	YES
123455	

BÀI D: SỐ ĐẸP

Một số được coi là đẹp nếu nó có tính chất thuận nghịch và tổng chữ số chia hết cho 10. Bài toán đặt ra là cho trước số chữ số. Hãy đếm xem có bao nhiêu số đẹp với số chữ số như vậy. **Dữ liệu vào:** Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng số chữ số tương ứng cần kiểm tra (lớn hơn 1 và nhỏ hơn 10)

Kết quả: Mỗi bộ test viết ra số lượng số đẹp tương ứng.

Input	Output
2	1
2	90
5	

BÀI E: SỐ THUẦN NGUYÊN TỐ

Một số được coi là thuần nguyên tố nếu nó là số nguyên tố, tất cả các chữ số là nguyên tố và tổng chữ số của nó cũng là một số nguyên tố. Bài toán đặt ra là đếm xem trong một đoạn giữa hai số nguyên cho trước có bao nhiêu số thuần nguyên tố.

Dữ liệu vào: Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng hai số nguyên dương tương ứng, cách nhau một khoảng trống. Các số đều không vượt quá 9 chữ số.

Kết quả: Mỗi bộ test viết ra số lượng các số thuần nguyên tố tương ứng.

Ví dụ

Input	Ouput
2	1
23 199	15
2345 6789	

BÀI F: MẢNG ĐỐI XỨNG

Nhập một dãy số nguyên có n phần tử (n không quá 100, các phần tử trong dãy không quá 10^9). Hãy viết chương trình kiểm tra xem dãy có phải đối xứng hay không. Nếu đúng in ra YES, nếu sai in ra NO. **Dữ liệu vào:** Dòng đầu ghi số bộ test, mỗi bộ test gồm hai dòng. Dòng đầu là số phần tử của dãy, dòng sau ghi ra dãy đó, mỗi số cách nhau một khoảng trống.

Input	Ouput
2	YES
4	NO
1 4 4 1	
5	
1 5 5 5 3	

BÀI G: SỐ TĂNG GIẢM

Một số được gọi là số tăng giảm nếu số đó có các chữ số thỏa mãn hoặc tăng dần, hoặc giảm dần từ trái qua phải. Hãy đếm các số nguyên tố là số tăng giảm với số chữ số cho trước. **Dữ liệu vào:** Dòng đầu tiên ghi số bộ test. Mỗi bộ test viết trên một dòng số chữ số tương ứng cần kiểm tra (lớn hơn 1 và nhỏ hơn 10)

Input	Output
2	20
2	50
4	

BÀI H: ĐẾM CÁC SỐ NGUYÊN TỐ TRONG DÃY

Cho dãy số A có n phần tử chỉ bao gồm các số nguyên dương (không quá 9 chữ số). Hãy xác định các số nguyên tố trong dãy và đếm xem mỗi số xuất hiện bao nhiêu lần.

Dữ liệu vào: Dòng đầu tiên ghi số bộ test. Với mỗi bộ test: dòng đầu ghi số n; dòng tiếp theo ghi n số của dãy.

Kết quả: Với mỗi bộ test ghi ra thứ tự bộ test, sau đó lần lượt là các số nguyên tố trong dãy **theo thứ tự từ nhỏ đến lớn** và số lần xuất hiện của nó.

Input	Output
1 10 1 2 3 3 2 1 3 2 7 8	Test 1: 2 xuất hiện 3 lần 3 xuất hiện 3 lần 7 xuất hiện 1 lần

BÀI I: CHÈN MẢNG

Nhập 2 mảng (a, N) và (b, M) và số nguyên p ($0 \leq p < M \leq N < 100$). Hãy chèn mảng b vào vị trí p của mảng a. **Dữ liệu vào:** Dòng đầu ghi số bộ test, mỗi bộ test gồm 3 dòng: dòng đầu ghi 3 số N,M,p. Dòng thứ 2 ghi N số của mảng a. Dòng thứ 3 ghi M số của mảng b. **Kết quả** ghi ra thứ tự bộ test và dãy số sau khi chèn.

Input	Output
1 4 3 1 5 3 6 7 2 9 11	Test 1: 5 2 9 11 3 6 7

BÀI J: ĐẾM SỐ PHẦN TỬ LỚN HƠN SỐ ĐỨNG TRƯỚC TRONG DÃY

Cho một dãy số nguyên dương có n phần tử ($2 \leq n \leq 50$). Hãy liệt kê số các phần tử trong dãy không nhỏ hơn các số đứng trước nó (tính cả phần tử đầu tiên).

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test ghi ra 2 dòng: dòng 1 là số phần tử, dòng thứ 2 ghi dãy số.

Kết quả ghi ra số phần tử thỏa mãn.

Input	Output
1 7 3 5 6 8 4 2 9	5

BÀI K: PHÂN TÍCH THỪA SỐ NGUYÊN TỐ

Hãy phân tích một số nguyên dương thành tích các thừa số nguyên tố.

Dữ liệu vào

Dòng đầu tiên ghi số bộ test.

Mỗi bộ test viết trên một dòng số nguyên dương n không quá 9 chữ số.

Kết quả

Mỗi bộ test viết ra thứ tự bộ test, sau đó lần lượt là các số nguyên tố khác nhau có trong tích, với mỗi số viết thêm số lượng số đó. Xem ví dụ để hiểu rõ hơn về cách viết kết quả.

Ví dụ

Input	Output
3	Test 1: 2 (2) 3 (1) 5 (1)
60	Test 2: 2 (7)
128	Test 3: 2 (4) 5 (4)
10000	

BÀI L: LIỆT KÊ VÀ ĐẾM 1

Cho một dãy rất nhiều các số nguyên dương, mỗi số cách nhau một khoảng trống. Hãy tìm các số thuận nghịch và đếm số lần xuất hiện của các số đó.

Dữ liệu vào: Gồm không quá 50000 số nguyên dương, các số đều không quá 9 chữ số.

Kết quả Ghi ra các số thuận nghịch kèm theo số lần xuất hiện. Các số được liệt kê theo thứ tự xuất hiện (tức là số nào xuất hiện trước in ra trước).

Ví dụ:

Input	Output
123 321 23456 123 123 23456 3523 123 321 4567 8988 878 7654 9899 3456 123 678 999 78 3456 878 987654321 4546 63543 4656 13432 4563 123471 659837 457745 34355 878 9087 9977 98534 3456 23132	878 3 999 1 457745 1 23132 1

BÀI M: LIỆT KÊ VÀ ĐẾM 2

Cho một dãy các số nguyên dương không quá 9 chữ số, mỗi số cách nhau một khoảng trống. Hãy tìm các số chỉ bao gồm các *chữ số nguyên tố* và đếm số lần xuất hiện của các số đó.

Dữ liệu vào: Gồm không quá 50000 số nguyên dương, các số đều không quá 9 chữ số.

Kết quả Ghi ra các số chỉ có các chữ số nguyên tố kèm theo số lần xuất hiện. Các số được liệt kê theo thứ tự xuất hiện (tức là số nào xuất hiện trước in ra trước).

Ví dụ:

Input	Output
123 321 23456 123 123 23456 3523 123 321 4567 8988 878 7654 9899 3456 123 678 999 77 3456 878 987654321 4546 63543 4656 13432 4563 123471 659837 57275 34355 878 9087 77 98534 3456 23132 3523 3523 3523	3523 4 77 2 57275 1

BÀI N: MA TRẬN XOÁY ỐC NGƯỢC

Ma trận xoáy ốc ngược cấp N là ma trận vuông có $N \times N$ phần tử. Các số được điền vào ma trận theo chiều kim đồng hồ theo thứ tự giảm dần về 1.

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test ghi số N ($1 < N < 20$).

Kết quả: Ghi ra thứ tự bộ test và ma trận xoáy ốc ngược tương ứng

Ví dụ:

<u>Input</u>	<u>Output</u>
1 3	Test 1: 9 8 7 2 1 6 3 4 5

BÀI O: MA TRẬN XOÁY ỐC NGUYÊN TỐ

Ma trận xoáy ốc nguyên tố cấp N là ma trận vuông có $N \times N$ phần tử. Các số được điền vào ma trận theo chiều kim đồng hồ đều là các số nguyên tố từ nhỏ đến lớn.

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test ghi số N ($1 < N < 15$).

Kết quả: Ghi ra thứ tự bộ test và ma trận xoáy ốc nguyên tố tương ứng

Ví dụ:

<u>Input</u>	<u>Output</u>
1 3	Test 1: 2 3 5 19 23 7 17 13 11

BÀI P: TÍCH MA TRẬN VỚI CHUYỂN VỊ CỦA NÓ

Cho ma trận A chỉ gồm các số nguyên dương cấp $N \times M$. Hãy viết chương trình tính tích của A với ma trận chuyển vị của A.

Dữ liệu vào: Dòng đầu tiên ghi số bộ test. Với mỗi bộ test: Dòng đầu tiên ghi hai số n và m là bậc của ma trận a; n dòng tiếp theo, mỗi dòng ghi m số của một dòng trong ma trận A.

Kết quả: Với mỗi bộ test ghi ra thứ tự bộ test, sau đó đến ma trận tích tương ứng, mỗi số cách nhau đúng một khoảng trống.

Ví dụ

Input	Output
1	Test 1:
2 2	5 11
1 2	11 25
3 4	

BÀI Q: LOẠI BỎ HÀNG VÀ CỘT CÓ TỔNG LỚN NHẤT RA KHỎI MA TRẬN

Cho một ma trận cấp $M \times N$ chỉ bao gồm các số nguyên dương không quá 1000. Hãy viết chương trình trước hết loại bỏ hàng có tổng lớn nhất sau đó tính toán lại để loại tiếp cột có tổng lớn nhất ra khỏi ma trận.

Dữ liệu vào Dòng đầu tiên ghi số bộ test.

Mỗi bộ test viết trên một dòng hai số N và M lần lượt là số hàng và số cột của ma trận ($2 < N, M < 10$).

Tiếp theo là N dòng ghi các số của ma trận.

Kết quả Mỗi bộ test viết ra thứ tự bộ test, sau đó là ma trận sau khi tính toán.

Ví dụ

Input	Output
1	Test 1:
3 3	1 4
1 2 4	3 0
3 4 0	
6 3 5	

BÀI R: XÓA HÀNG XÓA CỘT RA KHỎI MA TRẬN

Cho một ma trận vuông cấp $N \times N$ ($10 < N < 200$) chỉ bao gồm các số nguyên dương không quá 10^9 . Cần thực hiện K lần ($1 < K < 100$), mỗi lần xóa đi một hàng và một cột ra khỏi ma trận. Hãy in ra ma trận sau cùng.

Dữ liệu vào: Dòng đầu ghi số N. Tiếp theo là N dòng, mỗi dòng ghi N số của ma trận. Tiếp theo là một dòng ghi số K. Tiếp theo là K dòng, mỗi dòng ghi hai số i, j là chỉ số hàng và cột cần xóa (i và j được đảm bảo luôn trong phạm vi của ma trận, chỉ số tính từ 0). Trước hết xóa hàng i, sau đó sẽ xóa cột j.

Kết quả: Ghi ra ma trận cuối cùng sau khi xóa.

Ví dụ:

<u>Input</u>	<u>Output</u>
4	2 5
3 5 6 8	5 6
1 2 5 7	
6 8 9 4	
1 5 6 2	
2	

0 0	
1 2	

BÀI S: ĐẾM SỐ TỪ TRONG XÂU

Một từ trong chuỗi ký tự được định nghĩa là một dãy ký tự liên tiếp không có khoảng trống. Hãy đếm số từ của chuỗi và in kết quả ra màn hình.

Dữ liệu vào: Dòng đầu ghi số bộ test. Mỗi bộ test có một chuỗi không quá 200 ký tự.

Kết quả: Ghi ra số từ đếm được.

Ví dụ:

Input	Output
2	5
Lap trinh C va C++	2
ACBDSDS kdfjdkgfdgkhfgjlfgdkgjfdgdgfdgfd	

BÀI T: 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.

Dữ liệu vào:

Dòng đầu tiên ghi số bộ test.

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

Kết quả:

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	

BÀI U: CHUẨN HÓA XÂU HỌ TÊN

Các cán bộ, giảng viên PTIT khi tham gia hội nghị quốc tế sẽ được viết lại chuỗi họ tên theo dạng chuẩn trong đó họ được viết sau cùng, phân tách với phần tên đệm và tên bởi dấu phẩy. Các chữ cái của họ đều viết hoa.

Cho trước các chuỗi họ tên (có thể không chuẩn). Hãy đưa về dạng chuẩn tương ứng.

Dữ liệu vào:

- Dòng 1 ghi số N là chuỗi họ tên trong danh sách
- N dòng tiếp theo ghi lần lượt các chuỗi họ tên (không quá 50 ký tự)

Kết quả: Ghi ra các chuỗi chuẩn.

Ví dụ:

Input	Output
4	Quang Vinh, NGUYEN

nGUYEn	quaNG	vInH	Thi Thu Huong, TRAN
tRan	thi THU	huOnG	Quoc Vinh, NGO
nGO	quoC	VINH	Tuan Anh, LE
lE	tuAn	aNH	

BÀI V: TẬP TỪ RIÊNG CỦA HAI XÂU

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 abc ab ab ab abcd ab abc aaa xyz ab zzz abc dd dd abc xyz dd ttt sas cdc	abcd aaa ab abc zzz

BÀI W: ĐỊA CHỈ EMAIL PTIT

Địa chỉ email của các cán bộ, giảng viên PTIT được tạo ra bằng cách viết đầy đủ tên và ghép với các chữ cái đầu của họ và tên đệm. Nếu có nhiều người cùng email thì từ người thứ 2 sẽ thêm số thứ tự vào email đó.

Cho trước các chuỗi họ tên (có thể không chuẩn). Hãy tạo ra các địa email tương ứng.

Dữ liệu vào: Dòng 1 ghi số N là chuỗi họ tên trong danh sách. N dòng tiếp theo ghi lần lượt các chuỗi họ tên (không quá 50 ký tự)

Kết quả: Ghi ra các email được tạo ra.

Ví dụ:

Input	Output
4 nGUYEn quaNG vInH tRan thi THU huOnG nGO quoC VINH lE tuAn aNH	vinhnq@ptit.edu.vn huongttt@ptit.edu.vn vinhnq2@ptit.edu.vn anhlt@ptit.edu.vn

BÀI X: XÓA TỪ TRONG XÂU

Cho trước một chuỗi ký tự S1 chỉ bao gồm các chữ cái và khoảng trống cùng một từ S2. Hãy tìm xem S2 có xuất hiện trong S1 hay không. Nếu có loại bỏ tất cả những lần xuất hiện của S2 trong S1. Chú ý: tìm S2 trong S1 theo kiểu không phân biệt chữ hoa chữ thường

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test ghi trên hai dòng: Dòng đầu ghi chuỗi ký tự S1, độ dài không quá 200. Dòng thứ 2 ghi từ S2 (không quá 20 ký tự)

Kết quả: Với mỗi bộ test ghi ra thứ tự bộ test và xâu kết quả sau khi đã xóa.

Ví dụ:

Input	Output
2 Abc ddd abdc aaa bbb abc ddD XY aBc ACHDNC XXXX YYYY ABC ABC XXXX XxXx	Test 1: ddd abdc aaa bbb ddD XY Test 2: ACHDNC YYYY ABC ABC

BÀI Y: SỐ LA MÃ

Bảng chữ số La Mã bao gồm các chữ cái với ý nghĩa I=1; V=5; X=10; L=50; C=100; D=500; M=1000. Một số quy tắc viết các số La Mã như sau:

- Tính từ trái sang phải giá trị của các chữ số và nhóm chữ số giảm dần.
- I chỉ có thể đứng trước V hoặc X, X chỉ có thể đứng trước L hoặc C, C chỉ có thể đứng trước D hoặc M.
- Các chữ cái I, X, C, M, không được lặp lại quá ba lần liên tiếp; các chữ cái V, L, D không được lặp lại quá một lần liên tiếp.

Bài toán đặt ra là cho một xâu ký tự mô tả **đúng** một số La Mã. Hãy tính giá trị thập phân của số đó

Dữ liệu vào: Dòng đầu ghi số bộ test. Mỗi bộ test ghi trên một dòng dãy ký tự số La Mã.

Kết quả: Với mỗi bộ test ghi ra kết quả tương ứng

Ví dụ:

Input	Output
3 XIX DC CD	19 600 400

BÀI Z: TRỘN XÂU

Cho hai xâu ký tự S1 và S2 với độ dài N và chỉ chứa các ký tự từ A đến H. Chúng ta thực hiện thao tác như sau:

- Bước đầu tiên tạo S12 bằng cách lấy các ký tự lần lượt trong S1 và S2 từ trái sang phải, lấy một ký tự trong S2 trước, sau đó đến 1 ký tự trong S1 và cứ như vậy. Ví dụ S1 = ABCHAD, S2= DEFDAC thì S12 = DAEBFCDHAACD
- Sau đó ta lại lấy nửa bên trái của S12 thành S1 mới, nửa bên phải thành S2 mới. Trong ví dụ trên S1 mới là DAEBFC, S2 mới là DHAACD. Rồi lại tiếp tục như vậy trong các bước tiếp theo.

Cho trước một xâu S có độ dài $2*N$. Bài toán đặt ra là liệu có thể tạo ra xâu S sau một số lần lặp hay không.

Dữ liệu vào: Có nhiều bộ test, mỗi bộ test có bốn dòng. Dòng đầu ghi số N không quá 100. Dòng thứ 2 ghi S1, dòng thứ 3 ghi S2. Dòng cuối ghi xâu S. Input kết thúc với một dòng ghi số 0.

Kết quả: Ghi ra số bước lặp cần thiết. Nếu không thể tìm được thì ghi ra -1.

Ví dụ

Input	Output
4 AHAH HAHA HHAAAAHH 3 CDE CDE EEDDCC 0	2 -1

