

ÔN TẬP – CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

BÀI 1. XÂU NHỊ PHÂN CÓ K BIT 1	5
BÀI 2. XÂU AB	5
BÀI 3. TỔ HỢP TIẾP THEO.....	5
BÀI 4. HOÁN VỊ KẾ TIẾP	6
BÀI 5. CHỌN SỐ TỪ MA TRẬN VUÔNG CẤP N	6
BÀI 6. SẮP XẾP QUÂN HẬU 1	7
BÀI 7. SẮP XẾP QUÂN HẬU 2	7
BÀI 8. SỐ NHỎ NHẤT CÓ N ƯỚC SỐ.....	7
BÀI 9. TÌM BỘI SỐ	8
BÀI 10. MÁY ATM	8
BÀI 11. XEM PHIM.....	8
BÀI 12. NGƯỜI DU LỊCH.....	9
BÀI 13. KÝ TỰ LẶP TRONG HAI XÂU LIÊN TIẾP.....	9
BÀI 14. LŨY THỪA.....	10
BÀI 15. TÌM KIẾM NHỊ PHÂN.....	10
BÀI 16. GẤP ĐÔI DÃY SỐ	11
BÀI 17. DÃY XÂU FIBONACI.....	11
BÀI 18. ĐẾM SỐ BÍT 1	12
BÀI 19. SỐ FIBONACCI THỨ N	12
BÀI 20. LŨY THỪA MA TRẬN	12
BÀI 21. DÃY SỐ TRIBONACCI.....	13
BÀI 22. CHIA HẾT CHO 2.....	13
BÀI 23. BẢNG HÌNH CHỮ NHẬT	13
BÀI 24. ĐỔI TIỀN	14
BÀI 25. SẮP XẾP CÔNG VIỆC.....	14
BÀI 26. SỐ MAY MẮN	15
BÀI 27. NỐI DÂY.....	15
BÀI 28. NHÀM CHỮ SỐ.....	15
BÀI 29. XÓA CHỮ SỐ	16
BÀI 30. XEM PHIM 2.....	16

BÀI 31. XÂU CON CHUNG DÀI NHẤT	16
BÀI 32. DÃY CON TĂNG DÀI NHẤT	17
BÀI 33. DÃY CON CÓ TỔNG BẰNG S	17
BÀI 34. DÃY CON DÀI NHẤT CÓ TỔNG CHIA HẾT CHO K	17
BÀI 35. TỔ HỢP $C(n, k)$	18
BÀI 36. XÂU CON ĐỐI XỨNG DÀI NHẤT	18
BÀI 37. BẬC THANG	18
BÀI 38. HÌNH VUÔNG LỚN NHẤT	19
BÀI 39. SỐ CÓ TỔNG CHỮ SỐ BẰNG K	19
BÀI 40. ĐƯỜNG ĐI NHỎ NHẤT	20
BÀI 41. SẮP XẾP ĐỔI CHỖ TRỰC TIẾP	20
BÀI 42. SẮP XẾP CHỌN	20
BÀI 43. SẮP XẾP CHÈN	21
BÀI 44. SẮP XẾP NỔI BỌT	21
BÀI 45. SẮP XẾP NHANH	21
BÀI 46. SẮP XẾP KHÔNG NHANH	22
BÀI 47. SẮP XẾP LẠI DẠI CON	22
BÀI 48. BRT	22
BÀI 49. TÌM KIẾM	23
BÀI 50. MUA CÀ PHÊ	23
BÀI 51. XẾP HÀNG	24
BÀI 52. TÌM KIẾM XÂU	24
BÀI 53. TỔNG ĐA THỨC	25
BÀI 54. TRÒ CHƠI VÒNG TRÒN	25
BÀI 55. NGẮN XẾP 1	25
BÀI 56. NGẮN XẾP 2	26
BÀI 57. BIỂU THỨC HẬU TỔ 1	26
BÀI 58. BIỂU THỨC HẬU TỔ 2	27
BÀI 59. DÃY NGOẶC ĐÚNG DÀI NHẤT	27
BÀI 60. KIỂM TRA DÃY NGOẶC ĐÚNG	27
BÀI 61. SỬA LẠI DÃY NGOẶC	28
BÀI 62. XÓA DẤU NGOẶC	28

BÀI 63. TÍNH TOÁN GIÁ TRỊ BIỂU THỨC	29
BÀI 64. PHẦN TỬ BÊN PHẢI ĐẦU TIÊN LỚN HƠN	29
BÀI 65. HÌNH CHỮ NHẬT LỚN NHẤT	29
BÀI 66. HÌNH CHỮ NHẬT 0-1	30
BÀI 67: SỐ THỨ TỰ DẤU NGOẶC	31
BÀI 68: PREFIX TO INFIX	31
BÀI 69: PREFIX TO POSTFIX	32
BÀI 70: POSTFIX TO PREFIX	32
BÀI 71: POSTFIX TO INFIX	33
BÀI 72: INFIX TO POSTFIX	33
BÀI 73: DƯ THỪA DẤU NGOẶC	33
BÀI 74. ĐẢO NGƯỢC	34
BÀI 75. CẤU TRÚC DỮ LIỆU HÀNG ĐỢI 1	34
BÀI 76. CẤU TRÚC DỮ LIỆU HÀNG ĐỢI 2	35
BÀI 77. HÀNG ĐỢI HAI ĐẦU (DEQUEUE)	35
BÀI 78. ĐƯỜNG NGUYÊN TỐ	36
BÀI 79. QUAY HÌNH VUÔNG	37
BÀI 80. DI CHUYỂN	37
BÀI 81. GIEO MÀM	38
BÀI 82. SỐ BDN	39
BÀI 83: GIÁ TRỊ NHỎ NHẤT CỦA XÂU	39
BÀI 84. SỐ NHỊ PHÂN	39
BÀI 85. BỘI SỐ CHỈ CÓ 0 VÀ 9	40
BÀI 86. SỐ BDN NHỎ NHẤT CHIA HẾT CHO N	40
BÀI 87. BIẾN ĐỔI S - T	40
BÀI 88. BIẾN ĐỔI VỀ 1	41
BÀI 89. CHUYỂN TỪ DANH SÁCH CẠNH SANG DANH SÁCH KÈ	41
BÀI 90. CHUYỂN TỪ DANH SÁCH KÈ SANG DANH SÁCH CẠNH	42
BÀI 91. CHUYỂN MA TRẬN KÈ SANG DANH SÁCH KÈ	42
BÀI 92. CHUYỂN DANH SÁCH KÈ SANG MA TRẬN KÈ	42
BÀI 93. ĐẾM SỐ AO	43
BÀI 94. TÌM ĐƯỜNG ĐI TRONG ĐỒ THỊ VÔ HƯỚNG	43

BÀI 95. KIỂM TRA ĐỒ THỊ CÓ PHẢI LÀ CÂY HAY KHÔNG	44
BÀI 96. ĐỒ THỊ HAI PHÍA	44
BÀI 97. SỐ LƯỢNG HÒN ĐẢO.....	45
BÀI 98. HỘP MẶT.....	46
BÀI 99. QUÂN MÃ	46
BÀI 100. THUẬT TOÁN BFS	47
BÀI 101. THUẬT TOÁN DFS	47
BÀI 102. THÀNH PHẦN LIÊN THÔNG - BFS.....	48
BÀI 103. THÀNH PHẦN LIÊN THÔNG -DFS	48
BÀI 104. ĐƯỜNG ĐI - BFS.....	49
BÀI 105. ĐƯỜNG ĐI - DFS	50
BÀI 106. CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN BFS	50
BÀI 107. CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN DFS	51
BÀI 108. ĐỈNH KHỚP CỦA ĐỒ THỊ.....	51
BÀI 109. CẠNH CẦU CỦA ĐỒ THỊ.....	52
BÀI 110. CÂY KHUNG NHỎ NHẤT	52
BÀI 111. NỐI ĐIỂM.....	53
BÀI 112. ĐƯỜNG ĐI NGẮN NHẤT 1	53
BÀI 113. ĐƯỜNG ĐI NGẮN NHẤT 2	54
BÀI 114. BẢNG SỐ	55
BÀI 115. CÂY NHỊ PHÂN TÌM KIẾM.....	55
BÀI 116. ĐẾM TỪ.....	56
BÀI 117. CÂN ĐĨA.....	56
BÀI 118. NODE LÁ.....	57
BÀI 119. ĐỘ SÂU CỦA CÂY	58
BÀI 120. NODE TRUNG GIAN	59
BÀI 121. DUYỆT THEO THỨ TỰ GIỮA	59

BÀI 1. XÂU NHỊ PHÂN CÓ K BIT 1

Hãy in ra tất cả các chuỗi nhị phân độ dài N, có K bit 1 theo thứ tự từ điển tăng dần.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 20$). Mỗi test gồm 2 số nguyên N, K ($1 \leq K \leq N \leq 16$).

Output: Với mỗi test, in ra đáp án tìm được, mỗi chuỗi in ra trên một dòng.

Ví dụ:

Input	Output
2	0011
4 2	0101
3 2	0110
	1001
	1010
	1100
	011
	101
	110

BÀI 2. XÂU AB

Một chuỗi ký tự $S = (s_1, s_2, \dots, s_n)$ được gọi là chuỗi AB độ dài n nếu với mọi $s_i \in S$ thì s_i hoặc là ký tự A hoặc s_i là ký tự B. Ví dụ chuỗi $S = \text{"ABABABAB"}$ là một chuỗi AB độ dài 8. Cho số tự nhiên N và số tự nhiên K ($1 \leq K < N \leq 15$ được nhập từ bàn phím), hãy viết chương trình liệt kê tất cả các chuỗi AB có độ dài N chứa **duy nhất** một dãy K ký tự A liên tiếp.

Dữ liệu vào chỉ có một dòng ghi hai số N và K.

Kết quả ghi ra màn hình theo khuôn dạng:

- Dòng đầu tiên ghi lại số các chuỗi AB thỏa mãn yêu cầu bài toán;
- Những dòng kế tiếp, mỗi dòng ghi lại một chuỗi AB thỏa mãn. Các chuỗi được ghi ra theo thứ tự từ điển.

Ví dụ:

INPUT	OUTPUT
5 3	5
	AAABA
	AAABB
	ABAAA
	BAAAB
	BBAAA

BÀI 3. TỔ HỢP TIẾP THEO

Cho số nguyên dương ($1 < N < 40$) và số nguyên dương $K < N$. Với 1 tổ hợp chập K phần tử của N, hãy cho biết tổ hợp tiếp theo sẽ có bao nhiêu phần tử mới. Nếu tổ hợp đã cho là cuối cùng thì kết quả là K.

Dữ liệu vào: Dòng đầu ghi số bộ test, không quá 20. Mỗi bộ test viết trên hai dòng

- Dòng 1: hai số nguyên dương N và K ($K < N$)
- Dòng 2 ghi K số của tổ hợp ban đầu. Theo đúng thứ tự tăng dần, không có số nào trùng nhau.

Kết quả: Với mỗi bộ dữ liệu in ra số lượng phần tử mới.

Ví dụ:

INPUT	OUTPUT
3	1
5 3	2
1 3 5	4
5 3	
1 4 5	
6 4	
3 4 5 6	

BÀI 4. HOÁN VỊ KẾ TIẾP

Hãy viết chương trình nhận vào một chuỗi (có thể khá dài) các ký tự số và đưa ra màn hình hoán vị kế tiếp của các ký tự số đó (với ý nghĩa là hoán vị có giá trị lớn hơn tiếp theo nếu ta coi chuỗi đó là một giá trị số nguyên). Chú ý: Các ký tự số trong dãy có thể trùng nhau.

Ví dụ: 123 -> 132

279134399742 -> 279134423799

Cũng có trường hợp sẽ không thể có hoán vị kế tiếp. Ví dụ như khi đầu vào là chuỗi 987.

Dữ liệu vào: Dòng đầu tiên ghi số nguyên t là số bộ test ($1 \leq t \leq 1000$). Mỗi bộ test có một dòng, đầu tiên là số thứ tự bộ test, một dấu cách, sau đó là chuỗi các ký tự số, tối đa 80 phần tử.

Kết quả: Với mỗi bộ test hãy đưa ra một dòng gồm thứ tự bộ test, một dấu cách, tiếp theo đó là hoán vị kế tiếp hoặc chuỗi "BIGGEST" nếu không có hoán vị kế tiếp.

Ví dụ:

INPUT	OUTPUT
3	1 132
1 123	2 279134423799
2 279134399742	3 BIGGEST
3 987	

BÀI 5. CHỌN SỐ TỪ MA TRẬN VUÔNG CẤP N

Cho ma trận vuông $C_{i,j}$ cấp N ($1 \leq i, j \leq N \leq 10$) gồm N^2 số tự nhiên và số tự nhiên K (các số trong ma trận không nhất thiết phải khác nhau và đều không quá 100, K không quá 10^4). Hãy viết chương trình lấy mỗi hàng, mỗi cột duy nhất một phần tử sao cho tổng các phần tử này đúng bằng K .

Dữ liệu vào: Dòng 1 ghi hai số N và K . N dòng tiếp theo ghi ma trận C .

Kết quả: dòng đầu ghi số cách tìm được. Mỗi dòng tiếp theo ghi một cách theo vị trí của số đó trong lần lượt từng hàng của ma trận. Xem ví dụ để hiểu rõ hơn.

Ví dụ:

INPUT	OUTPUT
3 10	2
2 4 3	1 3 2
1 3 6	3 2 1
4 2 4	

BÀI 6. SẮP XẾP QUÂN HẬU 1

Cho một bàn cờ vua có kích thước $n * n$, ta biết rằng quân hậu có thể di chuyển theo chiều ngang, dọc, chéo. Vấn đề đặt ra rằng, có n quân hậu, bạn cần đếm số cách đặt n quân hậu này lên bàn cờ sao cho với 2 quân hậu bất kì, chúng không “ăn” nhau.

Input: Một số nguyên dương n duy nhất (không quá 10)

Output: Số cách đặt quân hậu.

Ví dụ:

Input	Output
4	2

BÀI 7. SẮP XẾP QUÂN HẬU 2

Cho một bàn cờ 8×8 , mỗi ô có một giá trị $A[i][j]$ nhất định ($0 \leq A[i][j] \leq 100$), tương ứng với điểm số đạt được nếu như bạn đặt một quân cờ vào đó.

Nhiệm vụ của bạn là đặt 8 quân hậu lên bàn cờ, sao cho không có 2 quân nào ăn nhau, và số điểm đạt được là lớn nhất.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 8 dòng, mỗi dòng 8 số nguyên mô tả bàn cờ.

Output: Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input	Output
1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64	260

BÀI 8. SỐ NHỎ NHẤT CÓ N ƯỚC SỐ

Cho số nguyên dương N . Nhiệm vụ của bạn là tìm số K nhỏ nhất, sao cho K có đúng N ước. Input đảm bảo rằng đáp án không vượt quá 10^{18} .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test gồm 1 số nguyên N ($1 \leq N \leq 1000$).

Output: Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input	Output
2	6
4	12
6	

BÀI 9. TÌM BỘI SỐ

Cho số nguyên N . Nhiệm vụ của bạn cần tìm số nguyên X nhỏ nhất là bội của N , và X chỉ chứa hai chữ số 0 và 9.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 10000$). Mỗi bộ test chứa số nguyên N trên một dòng ($1 \leq N \leq 500$).

Output: Với mỗi test in ra đáp án tìm được trên một dòng.

Ví dụ:

Input	Output
3	90
2	90
5	99
11	

BÀI 10. MÁY ATM

Một máy ATM hiện có n ($n \leq 30$) tờ tiền có giá trị $t[1], t[2], \dots, t[n]$. Hãy tìm cách trả ít tờ nhất với số tiền đúng bằng S (các tờ tiền có giá trị bất kỳ và có thể bằng nhau).

Input: Dòng đầu tiên gồm 2 số nguyên n và S ($S \leq 10^9$). Dòng thứ hai chứa n số nguyên $t[1], t[2], \dots, t[n]$ ($t[i] \leq 10^9$)

Output: Số tờ tiền ít nhất phải trả.

Ví dụ

Input	Output
3 5 1 4 5	1

BÀI 11. XEM PHIM

Nông dân John đang đưa các con bò của anh ta đi xem phim. Xe tải của anh ta thì có sức chứa tối đa là C ($100 \leq C \leq 7000$) kg, anh ta muốn đưa 1 số con bò đi xem phim sao cho tổng khối lượng của những con bò này là lớn nhất, đồng thời xe tải của anh ta vẫn còn có thể chở được. Cho N ($1 \leq N \leq 25$) con bò và khối lượng W_i của từng con, hãy cho biết khối lượng bò lớn nhất mà John có thể đưa đi xem phim là bao nhiêu.

Dữ liệu vào:

Dòng 1: 2 số nguyên cách nhau bởi dấu cách: C và N

Dòng 2.. $N+1$: Dòng $i+1$ chứa 1 số nguyên: W_i

Kết quả

Một số nguyên là tổng khối lượng bò lớn nhất mà John có thể mang đi xem phim.

Ví dụ:

Input	Output
259 5 81 58 42	242

33	
61	

BÀI 12. NGƯỜI DU LỊCH

Cho n thành phố đánh số từ 1 đến n và các tuyến đường giao thông hai chiều giữa chúng, mạng lưới giao thông này được cho bởi mảng $C[1 \dots n, 1 \dots n]$ ở đây $C[i][j] = C[j][i]$ là chi phí đi đoạn đường trực tiếp từ thành phố i đến thành phố j .

Một người du lịch xuất phát từ thành phố 1, muốn đi thăm tất cả các thành phố còn lại mỗi thành phố đúng 1 lần và cuối cùng quay lại thành phố 1. Hãy chỉ ra chi phí ít nhất mà người đó phải bỏ ra.

Dữ liệu vào: Dòng đầu tiên là số nguyên n – số thành phố ($n \leq 15$); n dòng sau, mỗi dòng chứa n số nguyên thể hiện cho mảng 2 chiều C .

Kết quả: Chi phí mà người đó phải bỏ ra.

Ví dụ:

INPUT	OUTPUT
4 0 20 35 10 20 0 90 50 35 90 0 12 10 50 12 0	117

BÀI 13. KÝ TỰ LẶP TRONG HAI XÂU LIÊN TIẾP

Cho một dãy các xâu ký tự chỉ bao gồm các chữ cái in hoa từ A đến Z, trong đó các ký tự trong mỗi xâu đều đã được sắp xếp theo thứ tự từ điển và mỗi chữ cái chỉ xuất hiện nhiều nhất một lần (tức là độ dài xâu tối đa là 26). Nếu một ký tự xuất hiện trong hai xâu liên tiếp thì được coi là một lần lặp. Hãy tìm cách sắp xếp lại thứ tự các xâu sao cho số lần lặp là nhỏ nhất có thể. Ví dụ dưới đây là cùng một dãy xâu nhưng với cách sắp xếp lại thì số lần lặp chỉ còn 2.

ABC		ABEF	
ABEF		DEF	
DEF	=> Số lần lặp là 6	ABC	=> Số lần lặp là 2.
ABCDE		FGH	
FGH		ABCDE	

Input: Dòng đầu tiên ghi số N ($2 \leq N \leq 10$) là số xâu ký tự. N dòng tiếp theo, mỗi dòng ghi một xâu.

Output: In ra trên một dòng số lần lặp nhỏ nhất có thể.

Ví dụ:

Test 1	Test 2	Test 3
Input : 5 ABC ABEF DEF ABCDE FGH Output : 2	Input : 6 BDE FGH DEF ABC BDE ABEF Output : 3	Input : 4 XYZ XYZ ABYZ Z Output : 4

BÀI 14. LŨY THỪA

Cho số nguyên dương N và K . Hãy tính N^K modulo 10^9+7 .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 1 số nguyên N và K ($1 \leq N \leq 1000$, $1 \leq K \leq 10^9$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input:	Output
2	8
2 3	1 6
4 2	

BÀI 15. TÌM KIẾM NHỊ PHÂN

Cho dãy số $A[]$ gồm có N phần tử đã được sắp xếp tăng dần và số K .

Nhiệm vụ của bạn là kiểm tra xem số K có xuất hiện trong dãy số hay không. Nếu có hãy in ra vị trí trong dãy $A[]$, nếu không in ra “NO”.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test bắt đầu bằng số nguyên N và K ($N \leq 100\,000$, $0 \leq K \leq 10^6$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($0 \leq A[i] \leq 10^6$), các phần tử là riêng biệt.

Output:

Với mỗi test in ra trên một dòng đáp án tìm được.

Ví dụ:

Input:	Output
2 5 3 1 2 3 4 5 6 5 0 1 2 3 9 10	3 NO

BÀI 16. GẤP ĐÔI DÃY SỐ

Một dãy số tự nhiên bắt đầu bởi con số 1 và được thực hiện N-1 phép biến đổi “gấp đôi” dãy số như sau:

Với dãy số A hiện tại, dãy số mới có dạng A, x, A trong đó x là số tự nhiên bé nhất chưa xuất hiện trong A.

Ví dụ với 2 bước biến đổi, ta có $[1] \rightarrow [1\ 2\ 1] \rightarrow [1\ 2\ 1\ 3\ 1\ 2\ 1]$.

Các bạn hãy xác định số thứ K trong dãy số cuối cùng là bao nhiêu?

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm số nguyên dương N và K ($1 \leq N \leq 50$, $1 \leq K \leq 2^N - 1$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Test ví dụ:

Input:	Output
2 3 2 4 8	2 4

Giải thích test 1: Dãy số thu được là [1, 2, 1, 3, 1, 2, 1].

Giải thích test 2: Dãy số thu được là [1, 2, 1, 3, 1, 2, 1, 4, 1, 2, 1, 3, 1, 2, 1].

BÀI 17. DÃY XÂU FIBONACI

Một dãy xâu ký tự G chỉ bao gồm các chữ cái A và B được gọi là dãy xâu Fibonacci nếu thỏa mãn tính chất: $G(1) = A$; $G(2) = B$; $G(n) = G(n-2) + G(n-1)$. Với phép cộng (+) là phép nối hai xâu với nhau. Bài toán đặt ra là tìm ký tự ở vị trí thứ i (tính từ 1) của xâu Fibonacci thứ n.

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test ghi trên một dòng 2 số nguyên N và i ($1 < N < 93$). Số i đảm bảo trong phạm vi của xâu G(N) và không quá 18 chữ số. **Kết quả:** Ghi ra màn hình kết quả tương ứng với từng bộ test.

Input	Output
2 6 4 8 19	A B

BÀI 18. ĐẾM SỐ BÍT 1

Cho số nguyên dương N . Mỗi bước, bạn sẽ biến đổi N thành $\lfloor N/2 \rfloor$, $N \bmod 2$, $\lfloor N/2 \rfloor$. Sau khi thực hiện một cách triệt để, ta thu được một dãy số chỉ toàn số 0 và 1.

Nhiệm vụ của bạn là hãy đếm các số bằng 1 trong đoạn $[L, R]$ của dãy số cuối cùng.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 3 số nguyên N, L, R ($1 \leq N, L, R < 2^{50}$, $0 \leq R-L \leq 100\,000$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input	Output
2	4
7 2 5	5
10 3 10	

Giải thích test 1: $[7] \rightarrow [3, 1, 3] \rightarrow [1, 1, 1, 1, 3] \rightarrow [1, 1, 1, 1, 1, 1, 1]$.

Giải thích test 2: Dãy số sau khi biến đổi là $[1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1]$.

BÀI 19. SỐ FIBONACCI THỨ N

Dãy số Fibonacci được xác định bằng công thức như sau:

$F[0] = 0, F[1] = 1;$

$F[n] = F[n-1] + F[n-2]$ với mọi $n \geq 2$.

Các phần tử đầu tiên của dãy số là 0, 1, 1, 2, 3, 5, 8, ...

Nhiệm vụ của bạn là hãy xác định số Fibonacci thứ n . Do đáp số có thể rất lớn, in ra kết quả theo modulo 10^9+7 .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 1000$).

Mỗi test bắt gồm một số nguyên N ($1 \leq N \leq 10^9$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input:	Output
3	1
2	8
6	6765
20	

BÀI 20. LŨY THỪA MA TRẬN

Cho ma trận vuông A kích thước $N \times N$. Nhiệm vụ của bạn là hãy tính ma trận $X = A^K$ với K là số nguyên cho trước. Đáp số có thể rất lớn, hãy in ra kết quả theo modulo 10^9+7 .

Input:

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

Mỗi test bắt gồm một số nguyên N và K ($1 \leq N \leq 10, 1 \leq K \leq 10^9$) là kích thước của ma trận và số mũ.

Output:

Với mỗi test, in ra kết quả của ma trận X.

Ví dụ:

Input:	Output
2	8 5
2 5	5 3
1 1	597240088 35500972 473761863
1 0	781257150 154135232 527013321
3 1000000000	965274212 272769492 580264779
1 2 3	
4 5 6	
7 8 9	

BÀI 21. DÃY SỐ TRIBONACCI

Dãy số Tribonacci được định nghĩa như sau:

$T[1] = 1; T[2] = 2; T[3] = 3;$

$T[i] = T[i-1] + T[i-2] + T[i-3]$ với mọi $i > 3$.

Đặt $F[N] = T[1] + T[2] + \dots + T[N]$. Nhiệm vụ của bạn là hãy tính $F[N]$ theo modulo $10^{15}+7$.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 100$). Mỗi test gồm một số nguyên N ($1 \leq N \leq 10^9$).

Output: Với mỗi test in ra đáp án tìm được trên một dòng.

Ví dụ:

Input	Output
5	1
1	3
2	6
3	12
4	23
5	

BÀI 22. CHIA HẾT CHO 2

Cho số nguyên dương N .

Nhiệm vụ của bạn là hãy xác định xem có bao nhiêu ước số của N chia hết cho 2?

Input:

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

Mỗi bộ test gồm một số nguyên N ($1 \leq N \leq 10^9$)

Output: Với mỗi test, in ra đáp án tìm được trên một dòng.

Test ví dụ:

Input	Output:
2	0
9	3
8	

BÀI 23. BẢNG HÌNH CHỮ NHẬT

Cho một bảng hình chữ nhật kích thước vô hạn. Ban đầu, tất cả các ô đều có giá trị bằng 0.

Có N phép thực hiện, mỗi bước, bạn được phép tăng giá trị của hình chữ nhật con từ ô (1, 1) tới ô (a, b) lên 1 đơn vị.

Sau N bước, số lớn nhất trong bảng là X. Nhiệm vụ của bạn là hãy đếm số lần xuất hiện của X?

Input:

Dòng đầu tiên là số nguyên N ($1 \leq N \leq 100$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên a và b mô tả một bước ($1 \leq a, b \leq 10^6$).

Output:

In ra số lần xuất hiện của số lớn nhất trong bảng.

Test ví dụ:

Input:	Output:
3 2 3 3 7 4 1	2

Giải thích test: Trạng thái cuối cùng của hình chữ nhật là:

```
1 0 0 0 0 0 0
2 1 1 1 1 1 1
3 2 2 1 1 1 1
3 2 2 1 1 1 1
```

BÀI 24. ĐỔI TIỀN

Chuẩn bị đi nước ngoài, Tí phải thực hiện đổi tiền. Tại ngân hàng có các mệnh giá bằng 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000. Tổng số tiền mà Tí mang đi đổi có giá trị bằng N.

Tí không muốn cầm nhiều tờ tiền. Các bạn hãy xác định xem Tí có ít nhất bao nhiêu tờ tiền sau khi đổi tiền?

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 50$).

Mỗi test gồm 1 số nguyên N ($1 \leq N \leq 100\,000$).

Output: Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input:	Output
2 70 121	2 3

BÀI 25. SẮP XẾP CÔNG VIỆC

Bạn được giao cho N công việc, công việc thứ i có thời gian bắt đầu là A[i] và kết thúc tại B[i]. Tại một thời điểm, bạn chỉ có thể làm một công việc.

Bạn hãy lựa chọn các công việc một cách tối ưu sao cho số công việc làm được là nhiều nhất.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test gồm 1 số nguyên N ($1 \leq N \leq 100\,000$).

N dòng tiếp theo, mỗi dòng gồm 2 số A[i] và B[i] ($0 \leq A[i] < B[i] \leq 10^6$).

Output: Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input	Output
1	4
6	
5 9	
1 2	
3 4	
0 6	
5 7	
8 9	

Giải thích test: Lựa chọn công việc 2, 3, 5, 6.

BÀI 26. SỐ MAY MẮN

Hoàng yêu thích các số may mắn. Ta biết rằng một số là *số may mắn* nếu biểu diễn thập phân của nó chỉ chứa các chữ số may mắn là 4 và 7. Ví dụ, các số 47, 744, 4 là số may mắn và 5, 17, 467 không phải. Hoàng muốn tìm số may mắn bé nhất có tổng các chữ số bằng n . Hãy giúp anh ấy

Dữ liệu vào: Dòng đầu ghi số bộ test, mỗi bộ test có một dòng chứa số nguyên n ($1 \leq n \leq 10^6$) — tổng các chữ số của số may mắn cần tìm.

Kết quả: In ra trên 1 dòng số may mắn bé nhất, mà tổng các chữ số bằng n . Nếu không tồn tại số thỏa mãn, in ra -1.

Ví dụ:

Input	Output
2	4 7
11	-1
10	

BÀI 27. NỐI DÂY

Có N sợi dây cần nối lại với nhau thành một sợi duy nhất. Mỗi lần chỉ được phép nối 2 sợi dây với nhau. Thời gian để nối hai sợi dây có độ dài a và b mất tổng cộng $a + b$ phút.

Hãy tính xem cần ít nhất bao nhiêu thời gian để có thể nối xong N sợi dây?

Dữ liệu vào: Dòng đầu tiên là số nguyên N ($N \leq 2 \cdot 10^6$). Dòng tiếp theo gồm N số nguyên dương $c[i]$ ($1 \leq c[i] \leq 10^9$).

Kết quả: In ra đáp án của bài toán theo modulo $10^9 + 7$.

Ví dụ:

Input:	Output
7	59
2 4 1 2 10 2 3	

BÀI 28. NHẦM CHỮ SỐ

Trong một buổi học toán, giáo viên viết 2 số nguyên, A và B , và yêu cầu Tèo thực hiện phép cộng. Tèo hông bao giờ tính toán sai, nhưng thỉnh thoảng cậu ta không chép các con số một cách chính xác. Lỗi duy nhất của là ghi nhầm '5' thành '6' hoặc ngược lại. Cho hai số, A và B , tính tổng nhỏ nhất và lớn nhất mà Tèo có thể nhận được.

Input: Có một dòng chứa hai số nguyên dương A và B ($1 \leq A, B \leq 1\,000\,000$).

Output: In ra 2 số nguyên cách nhau một dấu cách, tổng nhỏ nhất và lớn nhất có thể nhận được.

Ví dụ:

Test 1	Test 2	Test 3
Input: 11 25 Output: 36 37	Input: 1430 4862 Output: 6282 6292	Input: 16796 58786 Output: 74580 85582

BÀI 29. XÓA CHỮ SỐ

Cho một số có N chữ số. Bạn hãy xóa đi K chữ số để được số còn lại sau khi xóa là lớn nhất có thể.

Input: Dòng 1: số N và K ($1 \leq K < N \leq 500\,000$).

Dòng 2: Số có N chữ số, bắt đầu bằng số khác 0.

Output: Số lớn nhất có thể sau khi xóa K chữ số.

Ví dụ

Input	Output
4 2 1924	94

BÀI 30. XEM PHIM 2

Vẫn là John và những con bò. Khi đàn bò ngày càng trở nên đông hơn, anh ấy quyết định mua xe tải to hơn với khả năng chở được C kg ($1000 \leq C \leq 25000$). Cho số con bò là N ($20 \leq N \leq 100$) và khối lượng W_i của từng con, hãy cho biết khối lượng bò lớn nhất mà John có thể đưa đi xem phim là bao nhiêu.

Dữ liệu vào:

Dòng 1: 2 số nguyên cách nhau bởi dấu cách: C và N

Dòng 2..N+1: Dòng i+1 chứa 1 số nguyên: W_i

Kết quả

Một số nguyên là tổng khối lượng bò lớn nhất mà John có thể mang đi xem phim.

Ví dụ:

Input:	Output
259 5 81 58 42 33 61	242

BÀI 31. XÂU CON CHUNG DÀI NHẤT

Xâu ký tự X được gọi là xâu con của xâu ký tự Y nếu ta có thể xóa đi một số ký tự trong xâu Y để được xâu X.

Cho hai chuỗi ký tự A và B dài không quá 1000 ký tự (chữ cái viết thường hoặc chữ số), hãy tìm chuỗi ký tự C có độ dài lớn nhất và là con của cả A và B.

Input: Dòng 1: chứa chuỗi A. Dòng 2: chứa chuỗi B

Output: Chỉ gồm một dòng ghi độ dài chuỗi C tìm được

Ví dụ:

Input	Output
abc1def2ghi3 abcdefghijkl23	10

BÀI 32. Dãy con tăng dài nhất

Cho một dãy số nguyên gồm N phần tử $A[1], A[2], \dots, A[N]$.

Biết rằng dãy con tăng đơn điệu là 1 dãy $A[i_1], \dots, A[i_k]$

thỏa mãn $i_1 < i_2 < \dots < i_k$ và $A[i_1] < A[i_2] < \dots < A[i_k]$.

Hãy cho biết dãy con tăng đơn điệu dài nhất của dãy này có bao nhiêu phần tử?

Input: Dòng 1 gồm 1 số nguyên là số N ($1 \leq N \leq 1000$). Dòng thứ 2 ghi N số nguyên $A[1], A[2], \dots, A[N]$ ($1 \leq A[i] \leq 10000$).

Output: Ghi ra độ dài của dãy con tăng đơn điệu dài nhất.

Ví dụ:

Input	Output
6 1 2 5 4 6 2	4

BÀI 33. Dãy con có tổng bằng S

Cho N số nguyên dương tạo thành dãy $A = \{A_1, A_2, \dots, A_N\}$. Tìm ra một dãy con của dãy A (không nhất thiết là các phần tử liên tiếp trong dãy) có tổng bằng S cho trước.

Input: Dòng đầu tiên ghi hai số nguyên dương N và S ($0 < N \leq 200$) và S ($0 < S \leq 40000$). Các dòng tiếp theo lần lượt ghi N số hạng của dãy A là các số A_1, A_2, \dots, A_N ($0 < A_i \leq 200$).

Output: Nếu bài toán vô nghiệm thì in ra “NO”. Nếu bài toán có nghiệm thì in ra “YES”

Ví dụ:

Input	Output
5 6 1 2 4 3 5	YES

BÀI 34. Dãy con dài nhất có tổng chia hết cho K

Cho một dãy gồm n ($n \leq 1000$) số nguyên dương A_1, A_2, \dots, A_n và số nguyên dương k ($k \leq 50$).

Hãy tìm dãy con gồm nhiều phần tử nhất của dãy đã cho sao cho tổng các phần tử của dãy con này chia hết cho k.

Input: Dòng đầu tiên chứa hai số n, k ghi cách nhau bởi ít nhất 1 dấu trống. Các dòng tiếp theo chứa các số A_1, A_2, \dots, A_n được ghi theo đúng thứ tự cách nhau ít nhất một dấu trống hoặc xuống dòng.

Output: Gồm 1 dòng duy nhất ghi số lượng phần tử của dãy con dài nhất thỏa mãn

Ví dụ:

Input	Output
10 3 2 3 5 7 9 6 12 7 11 15	9

BÀI 35. TỔ HỢP $C(n, k)$

Cho 2 số nguyên n, k . Bạn hãy tính $C(n, k)$ modulo 10^9+7 .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 2 số nguyên n, k ($1 \leq k \leq n \leq 1000$).

Output:

Với mỗi test, in ra đáp án trên một dòng.

Ví dụ:

Input	Output
2	10
5 2	120
10 3	

BÀI 36. XÂU CON ĐỐI XỨNG DÀI NHẤT

Cho chuỗi S chỉ bao gồm các ký tự viết thường và dài không quá 5000 ký tự.

Hãy tìm chuỗi con đối xứng dài nhất của S .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test gồm một chuỗi S có độ dài không vượt quá 5000, chỉ gồm các ký tự thường.

Output:

Với mỗi test, in ra đáp án tìm được.

Ví dụ:

Input	Output
2	5
abcbadd	5
aaaaa	

BÀI 37. BẬC THANG

Một chiếc cầu thang có N bậc. Mỗi bước, bạn được phép bước lên trên tối đa K bậc. Hỏi có tất cả bao nhiêu cách bước để đi hết cầu thang? (Tổng số bước đúng bằng N).

Input:

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

Mỗi test gồm hai số nguyên dương N và K ($1 \leq N \leq 100000$, $1 \leq K \leq 100$).

Output:

Với mỗi test, in ra đáp án tìm được trên một dòng theo modulo 10^9+7 .

Ví dụ:

Input	Output
2	2
2 2	5
4 2	

Giải thích test 1: Có 2 cách đó là (1, 1) và (2).

Giải thích test 2: 5 cách đó là: (1, 1, 1, 1), (1, 1, 2), (1, 2, 1), (2, 1, 1), (2, 2).

BÀI 38. HÌNH VUÔNG LỚN NHẤT

Cho một bảng số N hàng, M cột chỉ gồm 0 và 1. Bạn hãy tìm hình vuông có kích thước lớn nhất, sao cho các số trong hình vuông toàn là số 1.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test bắt đầu bởi 2 số nguyên N, M ($1 \leq N, M \leq 500$).

N dòng tiếp theo, mỗi dòng gồm M số mô tả một hàng của bảng.

Output:

Với mỗi test, in ra đáp án là kích thước của hình vuông lớn nhất tìm được trên một dòng.

Test ví dụ:

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

BÀI 39. SỐ CÓ TỔNG CHỮ SỐ BẰNG K

Cho 2 số nguyên N và K. Bạn hãy đếm số lượng các số có N chữ số mà tổng các chữ số của nó bằng K.

Lưu ý, chữ số 0 ở đầu không được chấp nhận.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 50$).

Mỗi test gồm 2 số nguyên N và K ($1 \leq N \leq 100, 0 \leq K \leq 50000$).

Output:

Với mỗi test, in ra đáp số tìm được theo modulo 10^9+7 trên một dòng.

Ví dụ:

Input:	Output
3	2
2 2	5
2 5	21
3 6	

Giải thích test 1: 11 và 20.

Giải thích test 2: 14, 23, 32, 41.

BÀI 40. ĐƯỜNG ĐI NHỎ NHẤT

Cho bảng $A[]$ kích thước $N \times M$ (N hàng, M cột). Bạn được phép đi sang trái, đi sang phải và đi xuống ô chéo dưới. Khi đi qua ô (i, j) , điểm nhận được bằng $A[i][j]$.

Hãy tìm đường đi từ ô $(1, 1)$ tới ô (N, M) sao cho tổng điểm là nhỏ nhất.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm số nguyên dương N và M .

N dòng tiếp theo, mỗi dòng gồm M số nguyên $A[i][j]$ ($0 \leq A[i] \leq 1000$).

Output:

Với mỗi test, in ra độ dài dãy con tăng dài nhất trên một dòng.

Ví dụ:

Input	Output
1	8
3 3	
1 2 3	
4 8 2	
1 5 3	

Giải thích test: Đường đi $(1, 1) \rightarrow (1, 2) \rightarrow (2, 3) \rightarrow (3, 3)$.

BÀI 41. SẮP XẾP ĐỔI CHỖ TRỰC TIẾP

Hãy thực hiện thuật toán sắp xếp đổi chỗ trực tiếp trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán. **Dữ liệu vào:** Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100). **Kết quả:** Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

Ví dụ:

Input	Output
4	Buoc 1: 2 7 5 3
5 7 3 2	Buoc 2: 2 3 7 5
	Buoc 3: 2 3 5 7

BÀI 42. SẮP XẾP CHỌN

Hãy thực hiện thuật toán sắp xếp chọn trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

Dữ liệu vào: Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

Kết quả: Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

Ví dụ:

Input	Output
4	Buoc 1: 2 7 3 5
5 7 3 2	Buoc 2: 2 3 7 5
	Buoc 3: 2 3 5 7

BÀI 43. SẮP XẾP CHÈN

Hãy thực hiện thuật toán sắp xếp chèn trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

Dữ liệu vào: Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

Kết quả: Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

Ví dụ:

Input	Output
4	Buoc 0: 5
5 7 3 2	Buoc 1: 5 7
	Buoc 2: 3 5 7
	Buoc 3: 2 3 5 7

BÀI 44. SẮP XẾP NỔI BỌT

Hãy thực hiện thuật toán sắp xếp nổi bọt trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

Dữ liệu vào: Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

Kết quả: Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

Ví dụ:

Input	Output
4	Buoc 1: 3 2 5 7
5 3 2 7	Buoc 2: 2 3 5 7

BÀI 45. SẮP XẾP NHANH

Cho dãy số A[] gồm có N phần tử. Nhiệm vụ của bạn là sắp xếp dãy số theo thứ tự tăng dần.

Dữ liệu vào: Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test bắt đầu bằng số nguyên N, ($N \leq 100\,000$)

Dòng tiếp theo gồm N số nguyên A[i] ($0 \leq A[i] \leq 10^6$).

Kết quả:

In ra các phần tử của dãy số sau khi được sắp xếp.

Ví dụ:

Input	Output
1	1 2 3 4 5
5	
2 4 1 3 5	

BÀI 46. SẮP XẾP KHÔNG NHANH

Cho dãy số $A[]$ gồm có N phần tử. Nhiệm vụ của bạn là sắp xếp dãy số theo thứ tự tăng dần.

Bộ test được xây dựng để bạn không thể AC nếu sử dụng các phiên bản của sắp xếp nhanh (Quick Sort).

Dữ liệu vào:

Mỗi test bắt đầu bằng số nguyên N , ($N \leq 100\,000$)

Dòng tiếp theo gồm N số nguyên $A[i]$ ($0 \leq A[i] \leq 10^{18}$).

Kết quả:

In ra các phần tử của dãy số sau khi được sắp xếp.

Ví dụ:

Input	Output
5 2 4 1 3 5	1 2 3 4 5

BÀI 47. SẮP XẾP LẠI DẠI CON

Cho dãy số $A[]$ gồm có N phần tử. Nhiệm vụ của bạn là hãy tìm một dãy con liên tiếp dài nhất, sao cho sau khi sắp xếp lại dãy con này, dãy số $A[]$ sẽ là một dãy số được sắp xếp tăng dần.

Dữ liệu vào:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test bắt đầu bằng số nguyên N ($N \leq 100\,000$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($0 \leq A[i] \leq 10^6$).

Kết quả:

Với mỗi test in ra độ dài dãy con tìm được trên một dòng. Nếu dãy đã cho đã được sắp xếp, in ra "YES".

Ví dụ:

Input	Output
3 9 0 1 15 25 6 7 30 40 50 5 1 2 4 3 5 4 1 2 3 4	4 2 YES

Giải thích test 1: Cần sắp xếp lại dãy con 15, 25, 6, 7.

BÀI 48. BRT

Thành phố X có N thị trấn trên trục đường chính. Tọa độ của các thị trấn lần lượt là $a[1]$, $a[2]$, ..., $a[N]$, các tọa độ này là phân biệt, không có 2 tọa độ nào trùng nhau.

Chính quyền thành phố muốn xây dựng một tuyến buýt nhanh BRT để kết nối 2 thị trấn gần nhau nhất với nhau.

Bạn hãy tính thử xem chiều dài của tuyến buýt này bằng bao nhiêu? Và có bao nhiêu cặp thị trấn có tiềm năng giống nhau để xây dựng tuyến BRT này.

Dữ liệu vào:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test bắt đầu bằng số nguyên N ($N \leq 100\,000$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($-10^9 \leq A[i] \leq 10^9$).

Kết quả:

Với mỗi test in ra 2 số nguyên C và D, lần lượt là khoảng cách ngắn nhất giữa 2 thị trấn, và số lượng cặp thị trấn có cùng khoảng cách ngắn nhất này.

Ví dụ:

Input	Output
2	2 1
4	2 2
6 -3 0 4	
3	
-2 0 2	

Giải thích test 2: Cặp thị trấn (1, 2) và (2, 3) có cùng khoảng cách.

BÀI 49. TÌM KIẾM

Cho dãy số $A[]$ gồm có N phần tử đã được sắp xếp tăng dần và số K.

Nhiệm vụ của bạn là kiểm tra xem số K có xuất hiện trong dãy số hay không. Nếu có hãy in ra vị trí trong dãy $A[]$, nếu không in ra “NO”.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).

Mỗi test bắt đầu bằng số nguyên N và K ($N \leq 10^6$, $0 \leq K \leq 10^6$).

Dòng tiếp theo gồm N số nguyên $A[i]$ ($0 \leq A[i] \leq 10^6$), các phần tử là riêng biệt.

Output: Với mỗi test in ra trên một dòng đáp án tìm được.

Ví dụ:

Input	Output
2	3
5 3	NO
1 2 3 4 5	
6 5	
0 1 2 3 9 10	

BÀI 50. MUA CÀ PHÊ

Tí rất thích uống cà phê đá, một thứ mà rất nhiều lập trình viên khác cũng yêu thích. Có N cửa hàng bán cà phê ở gần chỗ làm việc của Tí, giá bán cho mỗi cốc cà phê tại đây lần lượt là $X[i]$.

Mỗi ngày, Tí có thể sử dụng số tiền bằng $Y[i]$ để mua cà phê. Các bạn hãy xác định xem Tí có thể lựa chọn bao nhiêu cửa hàng để mua được cà phê cho mình?

Input:

Dòng đầu tiên số nguyên N ($1 \leq N \leq 100\,000$).

Dòng tiếp theo gồm N số nguyên $X[i]$ ($0 \leq X[i] \leq 10^6$), là giá bán cà phê của cửa hàng thứ i.

Dòng tiếp gồm số nguyên Q, là số ngày mà Tí muốn uống cà phê.

Q dòng cuối, mỗi dòng gồm một số nguyên $Y[i]$ ($0 \leq Y[i] \leq 10^6$), là số tiền mà Tí có thể dùng trong ngày thứ i.

Output:

Với mỗi ngày, in ra số cửa hàng mà Tí có thể mua được cà phê.

Ví dụ:

Input:	Output
5	0
3 10 8 6 11	4
4	1
1	5
10	
3	
11	

Giải thích test 1: Ngày 1, Tí không đủ tiền để mua cà phê.

BÀI 51. XẾP HÀNG

Tại sân bay, mọi người đang làm thủ tục để check in. Có tất cả N vị khách. Vị khách thứ i tới làm thủ tục tại thời điểm $T[i]$ và cần $D[i]$ thời gian để check in xong.

Các bạn hãy xác định xem thời điểm nào tất cả các vị khách làm xong thủ tục để lên máy bay?

Input

Dòng đầu tiên là số nguyên dương N ($N \leq 100$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên cho biết thời điểm đến của vị khách thứ i và thời gian vị khách này làm xong thủ tục check in. Các giá trị này không vượt quá 10^6 .

Output

In ra đáp án tìm được.

Ví dụ:

Input	Output
3	15
2 1	
8 3	
5 7	

BÀI 52. TÌM KIẾM XÂU

Cho 2 chuỗi P và T . Nhiệm vụ của bạn là hãy kiểm tra chuỗi T có xuất hiện trong chuỗi P hay không?

Các thuật toán có thể sử dụng: KMP, Hashing, Z-algorithm, ...

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 2 dòng, lần lượt là chuỗi P và T , có độ dài không vượt quá 10^6 .

Output:

Với mỗi test, in ra "YES" nếu chuỗi T xuất hiện trong chuỗi P , in ra "NO" trong trường hợp ngược lại.

Ví dụ:

Input:	Output
4	YES
abcdeff	NO
deff	YES
computer	YES
muter	
stringmatchingmat	
ingmat	

aaaaa	
aa	

BÀI 53. TỔNG ĐA THỨC

Cho hai đa thức có bậc không quá 10000 (chỉ viết ra các phần tử có hệ số khác 0). Hãy sử dụng danh sách liên kết đơn để viết chương trình tính tổng hai đa thức đó.

Dữ liệu vào: Dòng đầu ghi số bộ test. Mỗi bộ test có hai dòng, mỗi dòng ghi một đa thức theo mẫu như trong ví dụ. Số phần tử của đa thức không quá 20.

Chú ý: Bậc của các hạng tử luôn theo thứ tự giảm dần, trong đa thức chỉ có phép cộng và luôn được viết đầy đủ hệ số + số mũ (kể cả mũ 0).

Kết quả: Ghi ra một dòng đa thức tổng tính được (theo mẫu như ví dụ)

Ví dụ:

Input	Output
1 3*x^8 + 7*x^2 + 4*x^0 11*x^6 + 9*x^2 + 2*x^1 + 3*x^0	3*x^8 + 11*x^6 + 16*x^2 + 2*x^1 + 7*x^0

BÀI 54. TRÒ CHƠI VÒNG TRÒN

Có N người đứng thành một vòng tròn. Mỗi người được đánh thứ tự từ 1 đến N theo chiều kim đồng hồ. Trò chơi như sau: Ban đầu một số nguyên M được chọn. Mọi người bắt đầu đếm liên tiếp từ 1 đến M, bắt đầu từ người thứ nhất theo chiều kim đồng hồ. Sau khi số M được đếm thì người tiếp theo sẽ bị loại khỏi vòng, và người tiếp theo người bị loại sẽ tiếp tục đếm từ 1. Trò chơi dừng lại khi chỉ còn duy nhất một người

Cho hai số nguyên N và M, hãy tìm ra thứ tự của người chơi cuối cùng trong trò chơi:

Input

Dòng đầu tiên gồm 1 số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T dòng, mỗi dòng gồm 2 số nguyên N, M ($1 \leq N \leq 5000$, $1 \leq M \leq 10000$).

Output

Với mỗi bộ test, in ra trên một dòng kết quả là thứ tự của người chơi cuối cùng.

Ví dụ

Input	Output
2 5 4 6 4	2 1

BÀI 55. NGĂN XẾP 1

Cho một ngăn xếp các số nguyên. Các thao tác được mô tả trong file văn bản gồm 3 lệnh: push, pop và show. Trong đó thao tác push kèm theo một giá trị cần thêm (không quá 1000). Hãy viết chương trình ghi ra kết quả của các lệnh show.

Dữ liệu vào: Gồm nhiều dòng, mỗi dòng chứa một lệnh push, pop hoặc show. Input đảm bảo số lượng phần tử trong stack khi nhiều nhất cũng không vượt quá 200.

Kết quả: Ghi ra màn hình các phần tử đang có trong stack theo thứ tự lưu trữ mỗi khi gặp lệnh show. Các số viết cách nhau đúng một khoảng trống. Nếu trong stack không còn gì thì in ra dòng "empty"

Ví dụ:

Input	Output
push 3	3 5
push 5	3 5 7
show	3
push 7	
show	
pop	
pop	
show	

BÀI 56. NGĂN XẾP 2

Yêu cầu bạn xây dựng một stack với các truy vấn sau đây:

“PUSH x”: Thêm phần tử x vào stack ($0 \leq x \leq 1000$).

“PRINT”: In ra phần tử đầu tiên của stack. Nếu stack rỗng, in ra “NONE”.

“POP”: Xóa phần tử đầu tiên của stack. Nếu stack rỗng, không làm gì cả.

Input:

Dòng đầu tiên là số lượng truy vấn Q ($Q \leq 100000$).

Mỗi truy vấn có dạng như trên.

Output:

Với mỗi truy vấn “PRINT”, hãy in ra phần tử đầu tiên của stack. Nếu stack rỗng, in ra “NONE”.

Ví dụ:

Input:	Output
9	1
PUSH 1	3
PUSH 2	NONE
POP	
PRINT	
PUSH 3	
PRINT	
POP	
POP	
PRINT	

BÀI 57. BIỂU THỨC HẬU TỔ 1

Hãy sử dụng ngăn xếp để thực hiện việc chuyển các biểu thức trung tố sang hậu tố

Dữ liệu vào: Dòng 1 ghi số N (không quá 20) là số biểu thức trung tố (đúng khuôn dạng) chỉ bao gồm các phép cộng, trừ, nhân, chia, các chữ cái thường từ a đến z và các dấu ngoặc đơn. N dòng tiếp theo mỗi dòng ghi một biểu thức.

Kết quả: Ghi ra màn hình các biểu thức hậu tố kết quả.

Ví dụ:

Input	Output
1	ab+cd+*
((a+b) * (c+d))	

BÀI 58. BIỂU THỨC HẬU TỔ 2

Cho một dãy biểu thức dạng trung tố có thể chưa đúng khuôn dạng với các phép toán cộng trừ nhân chia, lũy thừa.

Nhiệm vụ của bạn là hãy chuyển biểu thức đã cho về dạng hậu tố.

Chú ý xem ví dụ để hiểu sự khác biệt với bài trước.

Input:

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

Mỗi dòng gồm một xâu S, không quá 50 kí tự. Các số trong biểu thức đại diện bởi một chữ cái thường.

Output:

Với mỗi test, in ra biểu thức hậu tố tìm được.

Ví dụ:

Input:	Output
2 a+b*c+d a+b*(c^d-e)^(f+g*h)-i	abc*+d+ abcd^e-fgh*+^*+i-

BÀI 59. DÃY NGOẶC ĐÚNG DÀI NHẤT

Cho một xâu chỉ gồm các kí tự '(' và ')'. Một dãy ngoặc đúng được định nghĩa như sau:

- Xâu rỗng là 1 dãy ngoặc đúng.
- Nếu A là 1 dãy ngoặc đúng thì (A) là 1 dãy ngoặc đúng.
- Nếu A và B là 2 dãy ngoặc đúng thì AB là 1 dãy ngoặc đúng.

Cho một xâu S. Nhiệm vụ của bạn là hãy tìm dãy ngoặc đúng dài nhất xuất hiện trong xâu đã cho.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm một xâu S có độ dài không vượt quá 10^5 kí tự.

Output: Với mỗi test in ra một số nguyên là độ dài dãy ngoặc đúng dài nhất tìm được.

Ví dụ:

Input:	Output
3 ((()) () () () (())))	2 4 6

BÀI 60. KIỂM TRA DÃY NGOẶC ĐÚNG

Cho một xâu chỉ gồm các kí tự '(', ')', '[', ']', '{', '}'. Một dãy ngoặc đúng được định nghĩa như sau:

- Xâu rỗng là 1 dãy ngoặc đúng.
- Nếu A là 1 dãy ngoặc đúng thì (A), [A], {A} là 1 dãy ngoặc đúng.
- Nếu A và B là 2 dãy ngoặc đúng thì AB là 1 dãy ngoặc đúng.

Cho một xâu S. Nhiệm vụ của bạn là xác định xâu S có là dãy ngoặc đúng hay không?

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 1 xâu S có độ dài không vượt quá 100 000.

Output:

Với mỗi test, in ra "YES" nếu như S là dãy ngoặc đúng, in ra "NO" trong trường hợp ngược lại.

Ví dụ:

Input:	Output
2	YES
[()] { } { [() ()] () }	NO
[(])	

BÀI 61. SỬA LẠI DÃY NGOẶC

Cho một chuỗi chỉ gồm các ký tự ‘(’, ‘)’ và có độ dài chẵn. Hãy đếm số lượng dấu ngoặc cần phải đổi chiều ít nhất, sao cho chuỗi mới thu được là một dãy ngoặc đúng.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 1 chuỗi S có độ dài không vượt quá 100 000, chỉ gồm dấu (và).

Output:

Với mỗi test, in ra đáp án tìm được trên một dòng.

Ví dụ:

Input:	Output
4	2
)) ((2
((((1
(((()	3
) (()) (((

BÀI 62. XÓA DẤU NGOẶC

Cho biểu thức toán học đúng, bạn cần tìm tất cả các biểu thức đúng có thể bằng cách xóa bỏ các cặp dấu ngoặc tương ứng với nhau từ biểu thức ban đầu.

Ví dụ: Cho biểu thức: $(2+(2*2)+2)$ Các biểu thức tìm được:

$(2+2*2+2)$

$2+(2*2)+2$

$2+2*2+2$

Các biểu thức $(2+2*2)+2$ và $2+(2*2+2)$ không được chấp nhận vì không xóa đi các cặp dấu ngoặc tương ứng với nhau

Dữ liệu vào: Một dòng chứa biểu thức gồm các số nguyên không âm, các dấu +, -, *, / và dấu ngoặc đơn.

Biểu thức không quá 200 ký tự, có chứa ít nhất 1 và không quá 10 cặp dấu ngoặc.

Kết quả: In ra tất cả các biểu thức khác nhau thỏa mãn đầu bài theo thứ tự từ điển

Ví dụ

Input	Output
$(1 + (2 * (3 + 4)))$	$(1 + (2 * 3 + 4))$ $(1 + 2 * (3 + 4))$ $(1 + 2 * 3 + 4)$ $1 + (2 * (3 + 4))$ $1 + (2 * 3 + 4)$ $1 + 2 * (3 + 4)$ $1 + 2 * 3 + 4$

BÀI 63. TÍNH TOÁN GIÁ TRỊ BIỂU THỨC

Cho biểu thức S với các toán tử +, -, *, / và dấu ngoặc (). Các toán hạng là các số có giá trị không vượt quá 100. Hãy tính giá trị biểu thức S. Phép chia thực hiện với số nguyên, input đảm bảo số bị chia luôn khác 0, đáp số biểu thức có không quá 10 chữ số.

Input:

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

Mỗi dòng gồm một xâu S, không quá 100 kí tự. Các toán hạng là các số nguyên không âm.

Output:

Với mỗi test, in ra đáp án tìm được.

Ví dụ:

Input	Output
4	16
$6*3+2-(6-4/2)$	2278
$100+99*22$	102
$6*(4*3)+5$	-1
$1-2$	

BÀI 64. PHẦN TỬ BÊN PHẢI ĐẦU TIÊN LỚN HƠN

Cho dãy số A[] gồm N phần tử. Với mỗi A[i], bạn cần tìm phần tử bên phải đầu tiên lớn hơn nó. Nếu không tồn tại, in ra -1.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên N ($1 \leq N \leq 100000$).

Dòng tiếp theo gồm N số nguyên A[i] ($0 \leq A[i] \leq 10^9$).

Output:

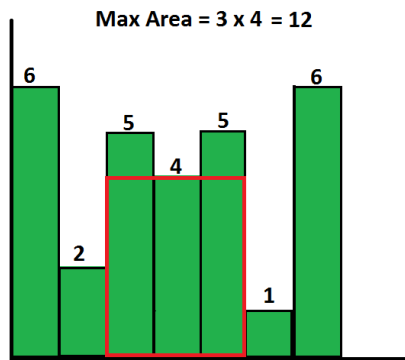
Với mỗi test, in ra trên một dòng N số R[i], với R[i] là giá trị phần tử đầu tiên lớn hơn A[i].

Ví dụ

Input	Output
3	5 25 25 -1
4	-1 -1 -1
4 5 2 25	5 5 -1 -1
3	
2 2 2	
4	
4 4 5 5	

BÀI 65. HÌNH CHỮ NHẬT LỚN NHẤT

Cho N cột, mỗi cột có chiều cao bằng H[i]. Bạn hãy tìm hình chữ nhật lớn nhất bị che phủ bởi các cột?



Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên N ($N \leq 100\,000$).

Dòng tiếp theo gồm N số nguyên $H[i]$ ($1 \leq H[i] \leq 10^9$).

Output:

Với mỗi test, in ra diện tích hình chữ nhật lớn nhất tìm được.

Ví dụ:

Input	Output
2	12
7	6
6 2 5 4 5 1 6	
3	
2 2 2	

BÀI 66. HÌNH CHỮ NHẬT 0-1

Cho một bảng kích thước $N \times M$, được chia thành lưới ô vuông đơn vị N dòng M cột. Trên các ô của bảng ghi số 0 hoặc 1. Các dòng của bảng được đánh số 1, 2... N theo thứ tự từ trên xuống dưới và các cột của bảng được đánh số 1, 2..., M theo thứ tự từ trái qua phải

Yêu cầu: Hãy tìm một hình chữ nhật gồm các ô của bảng thoả mãn các điều kiện sau:

- 1 - Hình chữ nhật đó chỉ gồm các số 1
- 2 - Cạnh hình chữ nhật song song với cạnh bảng
- 3 - Diện tích hình chữ nhật là lớn nhất có thể

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi hai số nguyên N và M ($1 \leq M, N \leq 500$).

N dòng tiếp theo, mỗi dòng gồm M số nguyên 0 hoặc 1.

Output:

Với mỗi test, in ra diện tích hình chữ nhật lớn nhất tìm được.

Ví dụ:

Input	Output
1 11 13 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 1 0 0 0 0 0 1 0 0 0 0 0 1 1	49

BÀI 67: SỐ THỨ TỰ DẤU NGOẶC

Cho biểu thức exp độ dài n chứa đựng một số ký tự ‘(,)’. Hãy in ra số thứ tự của các cặp ‘(,)’ khi phân tích biểu thức.

Input

Gồm nhiều dòng, mỗi dòng gồm một biểu thức exp ($1 \leq |\text{exp}| \leq 10^5$). Các dấu ngoặc trong biểu thức đảm bảo tuân thủ đúng quy tắc về đóng mở ngoặc.

Output

Mỗi dòng in ra số thứ tự của các cặp ‘(,)’ khi phân tích biểu thức tương ứng.

Ví dụ

Input	Output
(a+(b*c))+(d/e)	1 2 2 1 3 3
((()))((()))	1 2 3 3 2 4 5 5 4 1

BÀI 68: PREFIX TO INFIX

Có ba dạng biểu diễn cho các biểu thức số học và logic:

Infix (trung tố): Biểu diễn biểu thức dưới dạng trung tố là phép biểu diễn biểu thức trong đó phép toán được đặt giữa hai toán hạng. Ví dụ $(A+B) * (C-D)$.

Prefix (tiền tố): Biểu diễn biểu thức dưới dạng tiền tố là phép biểu diễn biểu thức trong đó phép toán được đặt trước hai toán hạng. Ví dụ $*+AB-CD$ (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Postfix (hậu tố): Biểu diễn biểu thức dưới dạng hậu tố là phép biểu diễn biểu thức trong đó phép toán được đặt sau hai toán hạng. Ví dụ $AB+CD-*$ (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng trung tố.

Input

Gồm nhiều dòng, mỗi dòng là một biểu thức tiền tố, trong đó các toán hạng là các chữ cái tiếng anh in thường.

Output

Mỗi dòng in ra một biểu thức trung tố tương ứng.

Ví dụ

Input	Output
*+ab-cd	((a+b) * (c-d))
*-a/bc-/akl	((a- (b/c)) * ((a/k) -1)

BÀI 69: PREFIX TO POSTFIX

Có ba dạng biểu diễn cho các biểu thức số học và logic:

Infix (trung tố): Biểu diễn biểu thức dưới dạng trung tố là phép biểu diễn biểu thức trong đó phép toán được đặt giữa hai toán hạng. Ví dụ $(A+B) * (C-D)$.

Prefix (tiền tố): Biểu diễn biểu thức dưới dạng tiền tố là phép biểu diễn biểu thức trong đó phép toán được đặt trước hai toán hạng. Ví dụ *+AB-CD (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Postfix (hậu tố): Biểu diễn biểu thức dưới dạng hậu tố là phép biểu diễn biểu thức trong đó phép toán được đặt sau hai toán hạng. Ví dụ AB+CD-* (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng hậu tố.

Input

Gồm nhiều dòng, mỗi dòng là một biểu thức tiền tố, trong đó các toán hạng là các chữ cái tiếng anh in thường.

Output

Mỗi dòng in ra một biểu thức hậu tố tương ứng.

Ví dụ

Input	Output
+ab-cd	ab+cd-
-a/bc-/akl	abc/-ak/l-

BÀI 70: POSTFIX TO PREFIX

Có ba dạng biểu diễn cho các biểu thức số học và logic:

Infix (trung tố): Biểu diễn biểu thức dưới dạng trung tố là phép biểu diễn biểu thức trong đó phép toán được đặt giữa hai toán hạng. Ví dụ $(A+B) * (C-D)$.

Prefix (tiền tố): Biểu diễn biểu thức dưới dạng tiền tố là phép biểu diễn biểu thức trong đó phép toán được đặt trước hai toán hạng. Ví dụ *+AB-CD (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Postfix (hậu tố): Biểu diễn biểu thức dưới dạng hậu tố là phép biểu diễn biểu thức trong đó phép toán được đặt sau hai toán hạng. Ví dụ AB+CD-* (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng hậu tố.

Input

Gồm nhiều dòng, mỗi dòng là một biểu thức hậu tố, trong đó các toán hạng là các chữ cái tiếng anh in thường.

Output

Mỗi dòng in ra một biểu thức tiền tố tương ứng.

Ví dụ

Input	Output
ab+cd-*	*+ab-cd
abc/-ak/l-*	*-a/bc-/akl

BÀI 71: POSTFIX TO INFIX

Có ba dạng biểu diễn cho các biểu thức số học và logic:

Infix (trung tố): Biểu diễn biểu thức dưới dạng trung tố là phép biểu diễn biểu thức trong đó phép toán được đặt giữa hai toán hạng. Ví dụ $(A+B) * (C-D)$.

Prefix (tiền tố): Biểu diễn biểu thức dưới dạng tiền tố là phép biểu diễn biểu thức trong đó phép toán được đặt trước hai toán hạng. Ví dụ $*+AB-CD$ (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Postfix (hậu tố): Biểu diễn biểu thức dưới dạng hậu tố là phép biểu diễn biểu thức trong đó phép toán được đặt sau hai toán hạng. Ví dụ $AB+CD-*$ (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng trung tố.

Input

Gồm nhiều dòng, mỗi dòng là một biểu thức hậu tố, trong đó các toán hạng là các chữ cái tiếng anh in thường.

Output

Mỗi dòng in ra một biểu thức trung tố tương ứng.

Ví dụ

Input	Output
abc++	$(a + (b + c))$
ab*c+	$((a * b) + c)$

BÀI 72: INFIX TO POSTFIX

Có ba dạng biểu diễn cho các biểu thức số học và logic:

Infix (trung tố): Biểu diễn biểu thức dưới dạng trung tố là phép biểu diễn biểu thức trong đó phép toán được đặt giữa hai toán hạng. Ví dụ $(A+B) * (C-D)$.

Prefix (tiền tố): Biểu diễn biểu thức dưới dạng tiền tố là phép biểu diễn biểu thức trong đó phép toán được đặt trước hai toán hạng. Ví dụ $*+AB-CD$ (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Postfix (hậu tố): Biểu diễn biểu thức dưới dạng hậu tố là phép biểu diễn biểu thức trong đó phép toán được đặt sau hai toán hạng. Ví dụ $AB+CD-*$ (tương ứng với biểu thức trung tố $(A+B)*(C-D)$).

Hãy viết chương trình chuyển đổi biểu thức biểu diễn dưới dạng tiền tố về dạng trung tố.

Input

Gồm nhiều dòng, mỗi dòng là một biểu thức trung tố, trong đó các toán hạng là các chữ cái tiếng anh in thường

Output

Mỗi dòng in ra một biểu thức hậu tố tương ứng

Ví dụ

Input	Output
$(a + (b + c))$	abc++
$((a * b) + c)$	ab*c+

BÀI 73: DƯ THỪA DẤU NGOẶC

Cho biểu thức số học, hãy cho biết biểu thức số học có dư thừa các cặp ký hiệu '(', ')' hay không?

Input

Gồm nhiều dòng, mỗi dòng là một biểu thức số học.

Output

Với mỗi biểu thức, nếu dư thừa các cặp kí hiệu ‘(’, ‘)’ thì in ra ‘Yes’, ngược lại in ra ‘No’.

Ví dụ

Input	Output
((a+b))	Yes
(a + (b) /c)	Yes
(a + b* (c-d))	No

BÀI 74. ĐẢO NGƯỢC

Cho một chuỗi ký tự bao gồm nhiều từ trong chuỗi. Hãy đảo ngược từng từ trong chuỗi.

Input

Gồm các chuỗi ký tự là các từ. Các ký tự của các từ là các chữ cái in hoa, in thường và các chữ số.

Output

Với mỗi từ in ra từ đảo ngược của nó. Các từ đảo ngược in ra trên một dòng.

Ví dụ

Input	Output
ABC Def 123	CBA feD 321 dBA
ABd	

BÀI 75. CẤU TRÚC DỮ LIỆU HÀNG ĐOỊ 1

Ban đầu cho một queue rỗng. Bạn cần thực hiện các truy vấn sau:

1. Trả về kích thước của queue
2. Kiểm tra xem queue có rỗng không, nếu có in ra “YES”, nếu không in ra “NO”.
3. Cho một số nguyên và đẩy số nguyên này vào cuối queue.
4. Loại bỏ phần tử ở đầu queue nếu queue không rỗng, nếu rỗng không cần thực hiện.
5. Trả về phần tử ở đầu queue, nếu queue rỗng in ra -1.
6. Trả về phần tử ở cuối queue, nếu queue rỗng in ra -1.

Dữ liệu vào

Dòng đầu tiên chứa số nguyên T là số bộ dữ liệu, mỗi bộ dữ liệu theo dạng sau.

Dòng đầu tiên chứa số nguyên n - lượng truy vấn ($1 \leq n \leq 1000$)

N dòng tiếp theo, mỗi dòng sẽ ghi loại truy vấn như trên, với truy vấn loại 3 sẽ có thêm một số nguyên, không quá 10^6 .

Kết quả: In ra kết quả của các truy vấn..

Ví dụ:

Input	Output
1	1
1 4	3
3 1	5
3 2	2
3 3	
5	
6	
4	
4	
4	
4	
4	
3 5	
3 6	
5	
1	

BÀI 76. CẤU TRÚC DỮ LIỆU HÀNG ĐỢI 2

Yêu cầu bạn xây dựng một queue với các truy vấn sau đây:

“PUSH x”: Thêm phần tử x vào cuối của queue ($0 \leq x \leq 1000$).

“PRINTFRONT”: In ra phần tử đầu tiên của queue. Nếu queue rỗng, in ra “NONE”.

“POP”: Xóa phần tử ở đầu của queue. Nếu queue rỗng, không làm gì cả.

Dữ liệu vào:

Dòng đầu tiên là số lượng truy vấn Q ($Q \leq 100000$).

Mỗi truy vấn có dạng như trên.

Kết quả:

Với mỗi truy vấn “PRINT”, hãy in ra phần tử đầu tiên của queue. Nếu queue rỗng, in ra “NONE”.

Ví dụ:

Input	Output
9	2
PUSH 1	2
PUSH 2	NONE
POP	
PRINTFRONT	
PUSH 3	
PRINTFRONT	
POP	
POP	
PRINTFRONT	

BÀI 77. HÀNG ĐỢI HAI ĐẦU (DEQUEUE)

Yêu cầu bạn xây dựng một hàng đợi hai đầu với các truy vấn sau đây:

“PUSHFRONT x”: Thêm phần tử x vào đầu của dequeue ($0 \leq x \leq 1000$).

“PRINTFRONT”: In ra phần tử đầu tiên của dequeue. Nếu dequeue rỗng, in ra “NONE”.

“POPFRONT”: Xóa phần tử đầu của dequeue. Nếu dequeue rỗng, không làm gì cả.

“PUSHBACK x”: Thêm phần tử x vào cuối của dequeue ($0 \leq x \leq 1000$).

“PRINTBACK”: In ra phần tử cuối của dequeue. Nếu dequeue rỗng, in ra “NONE”.

“POPBACK”: Xóa phần tử cuối của dequeue. Nếu dequeue rỗng, không làm gì cả.

Dữ liệu vào:

Dòng đầu tiên là số lượng truy vấn Q ($Q \leq 100000$).

Mỗi truy vấn có dạng như trên.

Kết quả:

Với mỗi truy vấn “PRINTFRONT” và “PRINTBACK”, hãy in ra kết quả trên một dòng.

Ví dụ:

Input	Output
10	2
PUSHBACK 1	1
PUSHFRONT 2	3
PUSHBACK 3	NONE
PRINTFRONT	
POPFRONT	
PRINTFRONT	
POPFRONT	
PRINTBACK	
POPFRONT	
PRINTBACK	

BÀI 78. ĐƯỜNG NGUYÊN TỐ

Cho hai số nguyên tố khác nhau có bốn chữ số. Người ta cho rằng hoàn toàn có thể biến đổi từ số này thành số kia sau một số bước theo quy tắc: Tại mỗi bước ta chỉ thay đổi một chữ số trong số trước đó sao cho số tạo được trong mỗi bước đều là một số nguyên tố có bốn chữ số. Một cách biến đổi như vậy gọi là một “đường nguyên tố”.

Bài toán đặt ra là với một cặp số nguyên tố đầu vào, hãy tính ra số bước của đường nguyên tố ngắn nhất. Giả sử đầu vào là hai số 1033 và 8179 thì đường nguyên tố ngắn nhất sẽ có độ dài là 6 với các bước chuyển là:

1033

1733

3733

3739

3779

8779

8179

Dữ liệu vào: Dòng đầu tiên ghi số bộ test, không lớn hơn 100. Mỗi bộ test viết trên một dòng bao gồm hai số nguyên tố có 4 chữ số..

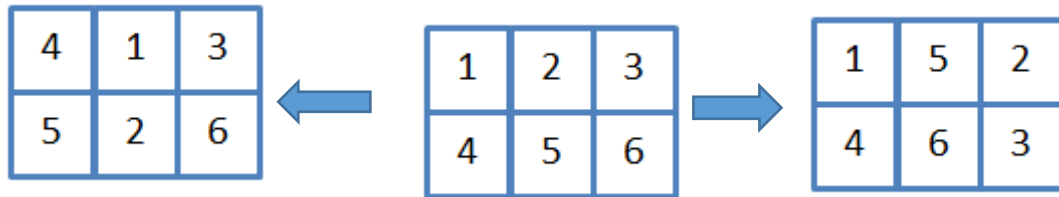
Kết quả: Với mỗi bộ test, in ra màn hình trên một dòng số bước của đường nguyên tố ngắn nhất.

Ví dụ:

Input	Output
3	6
1033 8179	7
1373 8017	0
1033 1033	

BÀI 79. QUAY HÌNH VUÔNG

Có một chiếc bảng hình chữ nhật với 6 miếng ghép, trên mỗi miếng ghép được điền một số nguyên trong khoảng từ 1 đến 6. Tại mỗi bước, chọn một hình vuông (bên trái hoặc bên phải), rồi quay theo chiều kim đồng hồ.



Yêu cầu: Cho một trạng thái của bảng, hãy tính số phép biến đổi ít nhất để đưa bảng đến trạng thái đích.

Dữ liệu vào:

Dòng đầu tiên chứa 6 số là trạng thái bảng ban đầu (thứ tự từ trái qua phải, dòng 1 tới dòng 2).

Dòng thứ hai chứa 6 số là trạng thái bảng đích (thứ tự từ trái qua phải, dòng 1 tới dòng 2).

Kết quả:

In ra một số nguyên là đáp số của bài toán.

Ví dụ:

Input	Output
1 2 3 4 5 6 4 1 2 6 5 3	2

BÀI 80. DI CHUYỂN

Cho một bảng kích thước $N \times N$, trong đó có các ô trống '.' và vật cản 'X'. Các hàng và các cột được đánh số từ 0.

Mỗi bước di chuyển, bạn có thể đi từ ô (x, y) tới ô (u, v) nếu như 2 ô này nằm trên cùng một hàng hoặc một cột, và không có vật cản nào ở giữa.

Cho điểm xuất phát và điểm đích. Bạn hãy tính số bước di chuyển ít nhất?

Dữ liệu vào:

Dòng đầu tiên là số nguyên dương N ($1 \leq N \leq 100$).

N dòng tiếp theo, mỗi dòng gồm N kí tự mô tả bảng.

Cuối cùng là 4 số nguyên a, b, c, d với (a, b) là tọa độ điểm xuất phát, (c, d) là tọa độ đích. Dữ liệu đảm bảo hai vị trí này không phải là ô cấm.

Kết quả:

In ra một số nguyên là đáp số của bài toán.

Ví dụ:

Input	Output
3 .X. .X. ... 0 0 0 2	3

BÀI 81. GIEO MÀM

Trên một giá có kích thước $R \times C$ (R hàng, C cột), một số hạt mầm đã được tra vào các ô. Một số hạt mầm được bón thêm chất dinh dưỡng, nên đã nảy mầm sớm thành cây non.

Mỗi ngày, các cây non sẽ lan truyền chất dinh dưỡng của nó cho các mầm ở ô xung quanh (trái, trên, phải, dưới), làm cho các hạt mầm này phát triển thành cây non. Tuy nhiên, có thể có một số hạt mầm được gieo ở vị trí lẻ loi, do không nhận được chất dinh dưỡng nên không thể nảy mầm. Các bạn hãy xác định xem cần ít nhất bao nhiêu ngày để tất cả các hạt đều mầm?

Dữ liệu vào:

Dòng đầu tiên gồm 2 số nguyên R và C ($1 \leq R, C \leq 500$).

R dòng tiếp theo, mỗi dòng gồm C số nguyên $A[i][j]$.

$A[i][j] = 0$, ô (i, j) là ô trống.

$A[i][j] = 1$, ô (i, j) là hạt chưa nảy mầm.

$A[i][j] = 2$, ô (i, j) là cây non.

Kết quả:

In ra thời gian ngắn nhất để tất cả các hạt đều nảy mầm. Nếu có hạt nào chưa nảy mầm, in ra -1.

Ví dụ:

Test 1	Test 2
Input: 3 5 2 1 0 2 1 1 0 1 2 1 1 0 0 2 1 Output: 2	Input: 3 5 2 1 0 2 1 0 0 1 2 1 1 0 0 2 1 Output: -1

BÀI 82. SỐ BDN

Ta gọi số nguyên dương K là một số BDN nếu các chữ số trong K chỉ bao gồm các 0 hoặc 1 có nghĩa. Ví dụ số $K = 1, 10, 101$. Cho số tự nhiên N ($N < 2^{63}$). Hãy cho biết có bao nhiêu số BDN không lớn hơn N . Ví dụ $N=100$ ta có 4 số BDN bao gồm các số: 1, 10, 11, 100.

Input	Output
3	2
10	4
100	7
200	

Dữ liệu vào: Dòng đầu tiên ghi lại số tự nhiên T ($T \leq 100$) là số lượng bộ test.

T dòng kế tiếp mỗi dòng là một số tự nhiên N ($1 \leq N \leq 2^{63}$).

Kết quả: Với mỗi số N in ra kết quả trên một dòng là số các số BDN không lớn hơn N .

BÀI 83: GIÁ TRỊ NHỎ NHẤT CỦA XÂU

Cho chuỗi ký tự $S[]$ bao gồm các ký tự in hoa ['A', 'B', ..., 'Z']. Ta định nghĩa giá trị của chuỗi $S[]$ là tổng bình phương số lần xuất hiện mỗi ký tự trong chuỗi. Ví dụ với chuỗi $S[] = \text{"AAABBCD"}$ ta có $F(S) = 3^2 + 2^2 + 1^2 + 1^2 = 15$. Hãy tìm giá trị nhỏ nhất của chuỗi $S[]$ sau khi loại bỏ K ký tự trong chuỗi.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 100$) là số lượng bộ test.

Mỗi bộ test có dạng như sau:

- Dòng thứ nhất gồm một số nguyên dương K ($1 \leq K \leq 10^5$).
- Dòng thứ hai gồm chuỗi ký tự S ($1 \leq |S| \leq 10^5$).

Output

Với mỗi bộ test in ra trên một dòng là giá trị nhỏ nhất của chuỗi S sau khi loại bỏ đi K ký tự trong chuỗi.

Ví dụ

Input	Output
2	2
2	3
ABCC	
1	
ABCC	

Giải thích

Ở bộ test đầu, ta bỏ đi 2 ký tự 'C'.

BÀI 84. SỐ NHỊ PHÂN

Cho số tự nhiên N . Hãy in ra tất cả các số nhị phân từ 1 đến N .

Input

Dòng đầu tiên ghi lại số lượng test T ($T \leq 100$).

Mỗi test là một số tự nhiên N được ghi trên một dòng ($N \leq 500$).

Output

Đưa ra kết quả mỗi test trên một dòng.

Ví dụ

Input	Output
2	1 10
2	1 10 11 100 101 110
5	

BÀI 85. BỘI SỐ CHỈ CÓ 0 VÀ 9

Cho số tự nhiên N . Hãy tìm số nguyên dương X nhỏ nhất được tạo bởi chữ số 9 và chữ số 0 và chia hết cho N . Ví dụ với $N = 5$ ta sẽ tìm ra $X = 90$.

Input

Dòng đầu tiên gồm một số nguyên dương T ($T \leq 100$) là số lượng bộ test.

Những dòng kế tiếp mỗi dòng ghi lại một test. Mỗi test là một số tự nhiên N được ghi trên một dòng ($N \leq 200$).

Output

Đưa ra theo từng dòng số X nhỏ nhất chia hết cho N tìm được.

Ví dụ

Input	Output
2	90
5	9009
7	

BÀI 86. SỐ BDN NHỎ NHẤT CHIA HẾT CHO N

Ta gọi số nguyên dương K là một số BDN nếu các chữ số trong K chỉ bao gồm các 0 hoặc 1 có nghĩa. Ví dụ số $K = 101$ là số BDN, $K = 102$ không phải là số BDN. Cho số tự nhiên N ($N < 1000$), hãy tìm số BDN nhỏ nhất của N .

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 100$) là số lượng bộ test.

T dòng tiếp theo gồm T bộ test, mỗi bộ test gồm 1 dòng là một số nguyên N ($1 \leq N \leq 1000$).

Output

Với mỗi bộ test in ra trên một dòng là số BDN nhỏ nhất chia hết cho N . Kết quả không đảm bảo nhỏ hơn 2^{63} .

Ví dụ

Input	Output
3	10
2	11100
12	11101
17	

BÀI 87. BIẾN ĐỔI S - T

Cho hai số nguyên dương S và T ($S, T \leq 10^4$) và hai thao tác (a), (b) dưới đây:

- Thao tác (a): Trừ S đi 1.
- Thao tác (b): Nhân S với 2.

Hãy dịch chuyển S thành T sao cho số lần thực hiện các thao tác (a), (b) là ít nhất. Ví dụ với $S = 2$, $T = 5$ thì số các bước ít nhất để dịch chuyển S thành T thông qua 4 thao tác sau:

- Thao tác (a): $2 \times 2 = 4$
- Thao tác (b): $4 - 1 = 3$
- Thao tác (a): $3 \times 2 = 6$
- Thao tác (b): $6 - 1 = 5$

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 100$) là số lượng bộ test.

T dòng tiếp theo, mỗi dòng gồm 2 số S, T ($S, T \leq 10^4$).

Output

Với mỗi bộ test, in ra số thao tác ít nhất để chuyển từ số S về số T.

Ví dụ

Input	Output
3	4
2 5	4
3 7	3
7 4	

BÀI 88. BIẾN ĐỔI VỀ 1

Cho số tự nhiên N ($N \leq 10^7$) và hai phép biến đổi (a), (b) dưới đây.

- Thao tác (a): Trừ N đi 1. Ví dụ N=17, thao tác (a) biến đổi N = 16.
- Thao tác (b): $N = \max(u, v)$ nếu $u \times v = N$ ($u, v > 1$). Ví dụ N=16, thao tác (b) có thể biến đổi $N = \max(2, 8) = 8$ hoặc $N = \max(4, 4) = 4$.

Chỉ được phép sử dụng hai thao tác (a) hoặc (b), hãy biến đổi N thành 1 sao số các thao tác (a), (b) được thực hiện ít nhất. Ví dụ với N=17, số các phép (a), (b) nhỏ nhất biến đổi N thành 1 là 4 bước như sau:

- Thao tác (a): $N = N - 1 = 16$
- Thao tác (b): $N = \max(4, 4) = 4$
- Thao tác (b): $N = \max(2, 2) = 2$
- Thao tác (a): $N = N - 1 = 1$

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 100$) là số lượng bộ test.

T dòng tiếp theo, mỗi dòng gồm một số nguyên N ($1 \leq N \leq 10^7$).

Output

Với mỗi bộ test, in ra trên một dòng số phép biến đổi ít nhất tìm được.

Ví dụ

Input	Output
3	4
17	5
50	5
100	

BÀI 89. CHUYỂN TỪ DANH SÁCH CẠNH SANG DANH SÁCH KÊ

Cho đơn đồ thị G vô hướng liên thông được mô tả bởi danh sách cạnh. Hãy in ra danh sách kề tương ứng của G.

Dữ liệu vào

- Dòng đầu tiên ghi hai số N và M là số đỉnh và số cạnh của đồ thị. $1 < N < M < 50$
- M dòng tiếp theo mỗi dòng ghi 1 cạnh lần lượt theo thứ tự từ nhỏ đến lớn của đồ thị

Kết quả

Ghi ra N dòng, mỗi dòng là danh sách kề của từng đỉnh theo thứ tự từ 1 đến N. Các danh sách được viết theo thứ tự tăng dần.

Ví dụ

Input	Output
3 3	2 3
1 2	1 3
1 3	1 2
2 3	

BÀI 90. CHUYỂN TỪ DANH SÁCH KỀ SANG DANH SÁCH CẠNH

Cho đơn đồ thị G vô hướng liên thông được mô tả bởi danh sách kề. Hãy in ra danh sách cạnh tương ứng của G.

Dữ liệu vào

- Dòng đầu tiên ghi số N là số đỉnh ($1 < N < 50$)
- N dòng tiếp theo mỗi dòng ghi 1 danh sách kề lần lượt theo thứ tự từ đỉnh 1 đến đỉnh N

Kết quả

Ghi ra lần lượt từng cạnh của đồ thị theo thứ tự tăng dần.

Ví dụ

Input	Output
3	1 2
2 3	1 3
1 3	2 3
1 2	

BÀI 91. CHUYỂN MA TRẬN KỀ SANG DANH SÁCH KỀ

Ma trận kề A của một đồ thị vô hướng là một ma trận chỉ có các số 0 hoặc 1 trong đó $A[i][j] = 1$ có ý nghĩa là đỉnh i kề với đỉnh j (chỉ số tính từ 1).

Danh sách kề thì liệt kê các đỉnh kề với đỉnh đó theo thứ tự tăng dần.

Hãy chuyển biểu diễn đồ thị từ dạng ma trận kề sang dạng danh sách kề.

Dữ liệu vào: Dòng đầu tiên chứa số nguyên n – số đỉnh của đồ thị ($1 < n \leq 1000$). n dòng tiếp theo, mỗi dòng có n số nguyên có giá trị 0 và 1 mô tả ma trận kề của đồ thị.

Kết quả: Gồm n dòng, dòng thứ i chứa các số nguyên là đỉnh có nối với đỉnh i và được sắp xếp tăng dần. Dữ liệu đảm bảo mỗi đỉnh có kết nối với ít nhất 1 đỉnh khác.

Input	Output
3	2 3
0 1 1	1 3
1 0 1	1 2
1 1 0	

BÀI 92. CHUYỂN DANH SÁCH KỀ SANG MA TRẬN KỀ

Cho đơn đồ thị vô hướng có n đỉnh dưới dạng danh sách kề.

Hãy biểu diễn đồ thị bằng ma trận kề.

Dữ liệu vào: Dòng đầu tiên chứa số nguyên n – số đỉnh của đồ thị ($1 \leq n \leq 1000$). n dòng tiếp theo, dòng thứ i chứa các số nguyên là các đỉnh kề với đỉnh i.

Kết quả: Ma trận kề của đồ thị.

Ví dụ:

Input	Output
3	0 1 1
2 3	1 0 1
1 3	1 1 0
1 2	

BÀI 93. ĐẾM SỐ AO

Sau khi thi trượt môn Cấu trúc dữ liệu và giải thuật, một số sinh viên D17 CNTT - PTIT quyết định bỏ học, đầu tư thuê đất để trồng rau. Mảnh đất thuê là một hình chữ nhật $N \times M$ ($1 \leq N \leq 100$; $1 \leq M \leq 100$) ô đất hình vuông. Nhưng chỉ sau đó vài ngày, trận lụt khủng khiếp đã diễn ra làm một số ô đất bị ngập. Mảnh đất bỗng biến thành các cái ao. Và sinh viên D17 lại dự định chuyển sang nuôi cá. Các bạn ấy muốn biết mảnh đất được chia thành bao nhiêu cái ao để có thể tính toán nuôi cá cho hợp lý. Hãy giúp các bạn ấy nhé. *Chú ý: Ao là gồm một số ô đất bị ngập có chung đỉnh. Để nhận thấy là một ô đất có thể có tối đa 8 ô chung đỉnh.*

Dữ liệu vào: Dòng 1: 2 số nguyên cách nhau bởi dấu cách: N và M . Dòng 2.. $N+1$: M kí tự liên tiếp nhau mỗi dòng đại diện cho 1 hàng các ô đất. Mỗi kí tự là 'W' hoặc '.' tương ứng với ô đất đã bị ngập và ô đất vẫn còn nguyên.

Kết quả: Một dòng chứa 1 số nguyên duy nhất là số ao tạo thành.

Input	Output
10 12 W.....WW. .WWW.....WWWWW....WW.WW.W.. ..W.....W.. .W.W.....WW. W.W.W.....W. .W.W.....W. ..W.....W.	3

BÀI 94. TÌM ĐƯỜNG ĐI TRONG ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị có N đỉnh và M cạnh. Có Q truy vấn, mỗi truy vấn yêu cầu trả lời câu hỏi giữa 2 đỉnh x và y có tồn tại đường đi tới nhau hay không?

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 2 số nguyên N, M ($1 \leq N, M \leq 1000$).

M dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v cho biết có cạnh nối giữa đỉnh u và v .

Dòng tiếp là số lượng truy vấn Q ($1 \leq Q \leq 1000$).

Q dòng tiếp theo, mỗi dòng gồm 2 số nguyên x và y .

Output:

Với mỗi truy vấn, in ra "YES" nếu có đường đi từ x tới y , in ra "NO" trong trường hợp ngược lại.

Ví dụ:

Input:	Output
1	NO
6 5	YES
1 2	
2 3	
3 4	
1 4	
5 6	
2	
1 5	
2 4	

BÀI 95. KIỂM TRA ĐỒ THỊ CÓ PHẢI LÀ CÂY HAY KHÔNG

Một đồ thị N đỉnh là một cây, nếu như nó có đúng $N-1$ cạnh và giữa 2 đỉnh bất kì, chỉ tồn tại duy nhất 1 đường đi giữa chúng.

Cho một đồ thị N đỉnh và $N-1$ cạnh, hãy kiểm tra đồ thị đã cho có phải là một cây hay không?

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên N ($1 \leq N \leq 1000$).

$N-1$ dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v cho biết có cạnh nối giữa đỉnh u và v .

Output:

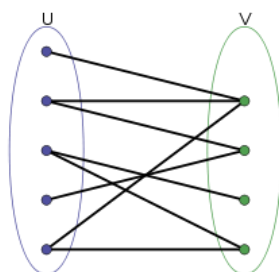
Với mỗi test, in ra “YES” nếu đồ thị đã cho là một cây, in ra “NO” trong trường hợp ngược lại.

Ví dụ:

Input	Output
2	YES
4	NO
1 2	
1 3	
2 4	
4	
1 2	
1 3	
2 3	

BÀI 96. ĐỒ THỊ HAI PHÍA

Đồ thị hai phía là một đồ thị đặc biệt, trong đó tập các đỉnh có thể được chia thành hai tập không giao nhau thỏa mãn điều kiện không có cạnh nối hai đỉnh bất kỳ thuộc cùng một tập. Cho đồ thị N đỉnh và M cạnh, bạn hãy kiểm tra đồ thị đã cho có phải là một đồ thị hai phía hay không?



Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên N và M ($1 \leq N, M \leq 1000$).

M dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v cho biết có cạnh nối giữa đỉnh u và v .

Output:

Với mỗi test, in ra “YES” nếu đồ thị đã cho là một đồ thị hai phía, in ra “NO” trong trường hợp ngược lại.

Ví dụ:

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

BÀI 97. SỐ LƯỢNG HÒN ĐẢO

Cho một bản đồ kích thước $N \times M$ được mô tả bằng ma trận $A[i][j]$. $A[i][j] = 1$ có nghĩa vị trí (i, j) là nổi trên biển. 2 vị trí (i, j) và (x, y) được coi là liên nhau nếu như nó có chung đỉnh hoặc chung cạnh. Một hòn đảo là một tập hợp các điểm (i, j) mà $A[i][j] = 1$ và có thể di chuyển giữa hai điểm bất kì trong đó.

Nhiệm vụ của bạn là hãy đếm số lượng đảo xuất hiện trên bản đồ.

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi 2 số nguyên N và M ($1 \leq N, M \leq 500$).

N dòng tiếp theo, mỗi dòng gồm M số nguyên $A[i][j]$.

Output: Với mỗi test, in ra số lượng hòn đảo tìm được.

Ví dụ:

Input:	Output
1	5
5 5	
1 1 0 0 0	
0 1 0 0 1	
1 0 0 1 1	
0 0 0 0 0	
1 0 1 0 1	

BÀI 98. HỌP MẶT

Có K người ($1 \leq K \leq 100$) đứng tại vị trí nào đó trong N địa điểm cho trước ($1 \leq N \leq 1,000$) được đánh số từ 1..N. Các điểm được nối với nhau bởi M đoạn đường một chiều ($1 \leq M \leq 10,000$) (không có đoạn đường nào nối một điểm với chính nó).

Mọi người muốn cùng tụ họp tại một địa điểm nào đó. Tuy nhiên, với các đường đi cho trước, chỉ có một số địa điểm nào đó có thể được chọn là điểm họp mặt. Cho trước K, N, M và vị trí ban đầu của K người cùng với M đường đi một chiều, hãy xác định xem có bao nhiêu điểm có thể được chọn làm điểm họp mặt.

Input

Dòng 1: Ghi 3 số: K, N, và M

Dòng 2 đến K+1: dòng i+1 chứa một số nguyên trong khoảng (1..N) cho biết địa điểm mà người thứ i đang đứng.

Dòng K+2 đến M+K+1: Mỗi dòng ghi một cặp số A và B mô tả một đoạn đường đi một chiều từ A đến B (cả hai trong khoảng 1..N và $A \neq B$).

Output

Số địa điểm có thể được chọn là điểm họp mặt.

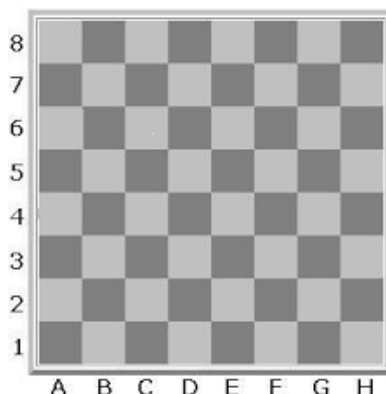
Ví dụ

Input:	Output:
2 4 4 2 3 1 2 1 4 2 3 3 4	2

Giải thích Ví dụ: có thể họp mặt tại điểm 3 và điểm 4.

BÀI 99. QUÂN MÃ

Cho một quân mã trên bàn cờ vua tại vị trí ST. Nhiệm vụ của bạn là hãy tìm số bước di chuyển ít nhất để đưa quân mã tới vị trí EN.



Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test gồm 2 xâu dạng “xy” và “uv”, trong đó x, y là kí tự trong “abcdefgh” còn v, v là số thuộc 1, 2, 3, 4, 5, 6, 7, 8.

Output:

Với mỗi test, in ra đáp án tìm được trên một dòng.

Ví dụ:

Input:	Output
8	2
e2 e4	4
a1 b2	2
b2 c3	6
a1 h8	5
a1 h7	6
h8 a1	1
b1 c3	0
f6 f6	

BÀI 100. THUẬT TOÁN BFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy thực hiện thuật toán duyệt đồ thị BFS bắt đầu tại một đỉnh $u \in V$.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 3 số nguyên $N=|V|$, $M=|E|$, u ($1 \leq N \leq 10^3$, $1 \leq M \leq 10^5$, $1 \leq u \leq N$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra trên một dòng theo thứ tự các đỉnh được duyệt trong quá trình duyệt đồ thị bằng thuật toán BFS bắt đầu tại đỉnh u .

Ví dụ

Input	Output
1	3 1 5 2 4
4 5 3	
1 2	
1 3	
2 4	
3 5	
4 5	

BÀI 101. THUẬT TOÁN DFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy thực hiện thuật toán duyệt đồ thị DFS bắt đầu tại một đỉnh $u \in V$.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 3 số nguyên $N=|V|$, $M=|E|$, u ($1 \leq N \leq 103$, $1 \leq M \leq 105$, $1 \leq u \leq N$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra trên một dòng theo thứ tự các đỉnh được duyệt trong quá trình duyệt đồ thị bằng thuật toán DFS bắt đầu tại đỉnh u .

Ví dụ

Input	Output
1 4 5 3 1 2 1 3 2 4 3 5 4 5	3 1 2 4 5

BÀI 102. THÀNH PHẦN LIÊN THÔNG - BFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy tìm tất cả các thành phần liên thông của đồ thị bằng thuật toán BFS.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 2 số nguyên $N=|V|$, $M=|E|$ ($1 \leq N \leq 103$, $1 \leq M \leq 105$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra trên nhiều dòng, mỗi dòng là các đỉnh thuộc cùng một thành phần liên thông, theo thứ tự duyệt BFS.

Ví dụ

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

BÀI 103. THÀNH PHẦN LIÊN THÔNG -DFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy tìm tất cả các thành phần liên thông của đồ thị bằng thuật toán DFS.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 2 số nguyên $N=|V|$, $M=|E|$ ($1 \leq N \leq 103$, $1 \leq M \leq 105$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra trên nhiều dòng, mỗi dòng là các đỉnh thuộc cùng một thành phần liên thông, theo thứ tự duyệt DFS.

Ví dụ

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

BÀI 104. ĐƯỜNG ĐI - BFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy tìm đường đi giữa hai đỉnh $u, v \in V$ bằng thuật toán BFS.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test. Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 4 số nguyên $N=|V|$, $M=|E|$, u, v ($1 \leq N \leq 103$, $1 \leq M \leq 105$, $1 \leq u, v \leq N$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra trên một dòng số lượng các cạnh trên đường đi từ u đến v theo thuật toán BFS. Nếu không có đường đi giữa u và v thì in ra -1.

Ví dụ

Input	Output
2 4 3 1 3 1 2 1 3 2 4 3 4 4 2 1 3 1 2 3 4	1 -1

BÀI 105. ĐƯỜNG ĐI - DFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy tìm đường đi giữa hai đỉnh $u, v \in V$ bằng thuật toán DFS.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 4 số nguyên $N=|V|$, $M=|E|$, u, v ($1 \leq N \leq 103$, $1 \leq M \leq 105$, $1 \leq u, v \leq N$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra trên một dòng số lượng các cạnh trên đường đi từ u đến v theo thuật toán DFS. Nếu không có đường đi giữa u và v thì in ra -1.

Ví dụ

Input	Output
2 4 3 1 3 1 2 1 3 2 4 3 4 4 2 1 3 1 2 3 4	3 -1

BÀI 106. CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN BFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy xây dựng một cây khung của đồ thị G với đỉnh $u \in V$ là gốc của cây bằng thuật toán BFS.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 3 số nguyên $N=|V|$, $M=|E|$, u ($1 \leq N \leq 103$, $1 \leq M \leq 105$, $1 \leq u \leq N$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, nếu tồn tại cây khung thì in ra $N - 1$ cạnh của cây khung với gốc là đỉnh u trên $N - 1$ dòng theo thứ tự duyệt của thuật toán BFS. Ngược lại nếu không tồn tại cây khung thì in ra -1.

Ví dụ

Input	Output
2 4 4 2 1 2 1 3	2 1 2 4 1 3 -1

2 4	
3 4	
4 2 2	
1 2	
3 4	

BÀI 107. CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN DFS

Cho đồ thị vô hướng $G=(V, E)$. Hãy xây dựng một cây khung của đồ thị G với đỉnh $u \in V$ là gốc của cây bằng thuật toán DFS.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 3 số nguyên $N=|V|$, $M=|E|$, u ($1 \leq N \leq 103$, $1 \leq M \leq 105$, $1 \leq u \leq N$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, nếu tồn tại cây khung thì in ra $N - 1$ cạnh của cây khung với gốc là đỉnh u trên $N - 1$ dòng theo thứ tự duyệt của thuật toán DFS. Ngược lại nếu không tồn tại cây khung thì in ra -1.

Ví dụ

Input	Output
2	2 1
4 3 2	1 3
1 2	3 4
1 3	-1
2 4	
3 4	
4 2 2	
1 2	
3 4	

BÀI 108. ĐỈNH KHỚP CỦA ĐỒ THỊ

Cho đồ thị vô hướng $G=(V, E)$. Hãy tìm tập các đỉnh khớp của đồ thị G . Đỉnh khớp của một đồ thị vô hướng là đỉnh mà nếu bỏ đi khỏi đồ thị cùng các cạnh kề với nó thì sẽ làm tăng số thành phần liên thông của đồ thị đó.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 2 số nguyên $N=|V|$, $M=|E|$ ($1 \leq N \leq 103$, $1 \leq M \leq 105$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra hai dòng.

Dòng đầu tiên gồm một số nguyên K ($K \geq 0$) là số lượng các đỉnh khớp của đồ thị.

Dòng thứ hai gồm K số nguyên u_1, u_2, \dots, u_K ($1 \leq u_1 < u_2 < \dots < u_K \leq N$) là số hiệu các đỉnh khớp của đồ thị.

Ví dụ

Input	Output
2	0
3 3	1
1 2	2
1 3	
2 3	
3 2	
1 2	
2 3	

BÀI 109. CẠNH CẦU CỦA ĐỒ THỊ

Cho đồ thị vô hướng $G=(V, E)$. Hãy tìm tập các cạnh cầu của đồ thị G . Cạnh cầu của một đồ thị vô hướng là cạnh mà nếu bỏ đi khỏi đồ thị thì sẽ làm tăng số thành phần liên thông của đồ thị đó.

Input

Dòng đầu tiên gồm một số nguyên T ($1 \leq T \leq 20$) là số lượng bộ test.

Tiếp theo là T bộ test, mỗi bộ test có dạng sau:

- Dòng đầu tiên gồm 2 số nguyên $N=|V|$, $M=|E|$ ($1 \leq N \leq 103$, $1 \leq M \leq 105$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên a, b ($1 \leq a, b \leq N$, $a \neq b$) tương ứng cạnh nối hai chiều từ a tới b .
- Dữ liệu đảm bảo giữa hai đỉnh chỉ tồn tại nhiều nhất một cạnh nối.

Output

Với mỗi bộ test, in ra hai dòng.

Dòng đầu tiên gồm một số nguyên K ($K \geq 0$) là số lượng các cạnh cầu của đồ thị.

Dòng thứ hai gồm K số nguyên u_1, u_2, \dots, u_K ($1 \leq u_1 < u_2 < \dots < u_K \leq M$) là số hiệu các cạnh cầu của đồ thị theo thứ tự đọc dữ liệu.

Ví dụ

Input	Output
2	0
3 2	
1 2	1
1 3	2
4 3	
1 2	
2 3	
3 4	

BÀI 110. CÂY KHUNG NHỎ NHẤT

Cho đơn đồ thị vô hướng liên thông $G = (V, E)$ gồm N đỉnh và M cạnh, các đỉnh được đánh số từ 1 tới N và các cạnh được đánh số từ 1 tới M . Hãy tìm cây khung nhỏ nhất của đồ thị G .

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên N và M . ($N, M \leq 200\,000$).

M dòng tiếp theo, mỗi dòng gồm 3 số nguyên u, v, c cho biết có cạnh nối giữa đỉnh u và v có trọng số bằng c ($1 \leq c \leq 10^6$).

Output:

Với mỗi test, in ra trọng số của cây khung nhỏ nhất tìm được.

Ví dụ:

Input:	Output
1 6 9 1 2 1 1 3 1 2 4 1 2 3 2 2 5 1 3 5 1 3 6 1 4 5 2 5 6 2	5

BÀI 111. NỐI ĐIỂM

Cho N điểm trên mặt phẳng Oxy. Để vẽ được đoạn thẳng nối A và B sẽ tốn chi phí tương đương với khoảng cách từ A tới B.

Nhiệm vụ của bạn là nối các điểm với nhau, sao cho N điểm đã cho tạo thành 1 thành phần liên thông duy nhất và chi phí để thực hiện là nhỏ nhất có thể.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi số nguyên N ($1 \leq N \leq 100$).

N dòng tiếp theo, mỗi dòng gồm 2 số thực x[i], y[i] là tọa độ của điểm thứ i ($|x[i]|, |y[i]| \leq 100$).

Output:

Với mỗi test, in ra chi phí nhỏ nhất tìm được với độ chính xác 6 chữ số thập phân sau dấu phẩy.

Ví dụ:

Input:	Output
1 3 1.0 1.0 2.0 2.0 2.0 4.0	3.414214

BÀI 112. ĐƯỜNG ĐI NGẮN NHẤT 1

Cho đơn đồ thị vô hướng liên thông $G = (V, E)$ gồm N đỉnh và M cạnh, các đỉnh được đánh số từ 1 tới N và các cạnh được đánh số từ 1 tới M.

Có Q truy vấn, mỗi truy vấn yêu cầu bạn tìm đường đi ngắn nhất giữa đỉnh X[i] tới Y[i].

Input:

Dòng đầu tiên hai số nguyên N và M ($1 \leq N \leq 100, 1 \leq M \leq N*(N-1)/2$).

M dòng tiếp theo, mỗi dòng gồm 3 số nguyên u, v, c cho biết có cạnh nối giữa đỉnh u và v có độ dài bằng c ($1 \leq c \leq 1000$).

Tiếp theo là số lượng truy vấn Q ($1 \leq Q \leq 100\,000$).

Q dòng tiếp theo, mỗi dòng gồm 2 số nguyên $X[i]$, $Y[i]$.

Output:

Với mỗi truy vấn, in ra đáp án là độ dài đường đi ngắn nhất tìm được.

Ví dụ:

Input:	Output
5 6	8
1 2 6	10
1 3 7	3
2 4 8	
3 4 9	
3 5 1	
4 5 2	
3	
1 5	
2 5	
4 3	

BÀI 113. ĐƯỜNG ĐI NGẮN NHẤT 2

Cho đồ thị vô hướng liên thông $G = (V, E)$ gồm N đỉnh và M cạnh, các đỉnh được đánh số từ 1 tới N và các cạnh được đánh số từ 1 tới M .

Nhiệm vụ của bạn là hãy tìm đường đi ngắn nhất từ 1 tới N và đếm xem có bao nhiêu tuyến đường có độ dài ngắn nhất như vậy?

Input:

Dòng đầu tiên hai số nguyên N và M ($1 \leq N \leq 10^5$, $1 \leq M \leq \max(N*(N-1)/2, 10^6)$).

M dòng tiếp theo, mỗi dòng gồm 3 số nguyên u , v , c cho biết có cạnh nối giữa đỉnh u và v có độ dài bằng c ($1 \leq c \leq 10^6$).

Output:

In ra 2 số nguyên là độ dài đường đi ngắn nhất và số lượng đường đi ngắn nhất. Input đảm bảo số lượng đường đi ngắn nhất không vượt quá 10^{18} .

Ví dụ:

Input:	Output
5 6	10 2
1 2 6	
1 3 7	
2 4 2	
3 4 9	
3 5 3	
4 5 2	

Có 2 tuyến đường ngắn nhất: $1 \rightarrow 3 \rightarrow 5$ và $1 \rightarrow 2 \rightarrow 4 \rightarrow 5$.

BÀI 114. BẢNG SỐ

Cho một bảng số kích thước $N \times M$. Chi phí khi đi qua ô (i, j) bằng $A[i][j]$. Nhiệm vụ của bạn là hãy tìm một đường đi từ ô $(1, 1)$ tới ô (N, M) sao cho chi phí là nhỏ nhất. Tại mỗi ô, bạn được phép đi sang trái, sang phải, đi lên trên và xuống dưới.

Input:

Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).

Mỗi test bắt đầu bởi hai số nguyên N và M ($1 \leq N, M \leq 500$).

N dòng tiếp theo, mỗi dòng gồm M số nguyên $A[i][j]$ ($0 \leq A[i][j] \leq 9$).

Output:

Với mỗi test, in ra một số nguyên là chi phí nhỏ nhất cho đường đi tìm được.

Ví dụ:

Input:	Output
3	24
4	15
5	13
0 3 1 2 9	
7 3 4 9 9	
1 7 5 5 3	
2 3 4 2 5	
1	
6	
0 1 2 3 4 5	
5 5	
1 1 1 9 9	
9 9 1 9 9	
1 1 1 9 9	
1 9 9 9 9	
1 1 1 1 1	

BÀI 115. CÂY NHỊ PHÂN TÌM KIẾM

Bạn hãy cài đặt cấu trúc cây nhị phân tìm kiếm (không cần đảm bảo tính chất cân bằng) với các node trên cây chứa các số nguyên dương không quá 1000 và thực hiện các loại truy vấn sau:

1. Dạng 1: Thêm một node vào cây
2. Dạng 2: Xóa một node trên cây và đưa node trái nhất của cây con phải lên thay thế
3. Dạng 3: Duyệt cây theo thứ tự trước và in ra
4. Dạng 4: Duyệt cây theo thứ tự giữa và in ra
5. Dạng 5: Duyệt cây theo thứ tự sau và in ra

Dữ liệu vào gồm nhiều dạng truy vấn, luôn bắt đầu bằng dạng 1, mỗi truy vấn trên một dòng.

Mỗi dòng sẽ ghi số thứ tự của dạng truy vấn. Trong đó truy vấn dạng 1 và 2 sẽ kèm theo một số nguyên dương không quá 1000 là số cần thêm hoặc loại ra khỏi cây (truy vấn dạng 1 được đảm bảo giá trị không trùng với các node đã có, truy vấn dạng 2 luôn đảm bảo chắc chắn đã có node cần loại ở trong cây). Cây được đảm bảo luôn có không quá 1000 node.

Kết quả in ra thứ tự duyệt khi gặp truy vấn 3,4,5.

Ví dụ:

Input	Output
1 30	30 16 10 4 17 57 40
1 15	4 10 16 17 30 40 57
1 50	4 10 17 16 40 57 30
1 10	
1 17	
1 40	
1 57	
1 68	
1 4	
1 16	
2 15	
2 50	
2 68	
3	
4	
5	

BÀI 116. ĐẾM TỪ

Cho luồng vào gồm không quá 10^6 từ, mỗi từ chỉ bao gồm các chữ cái viết hoa không dấu và có độ dài không quá 20 ký tự. Các từ phân cách bởi các khoảng trống hoặc dấu xuống dòng.

Hãy đếm xem có bao nhiêu từ khác nhau, mỗi từ xuất hiện bao nhiêu lần.

Kết quả được liệt kê theo thứ tự từ điển.

Ví dụ:

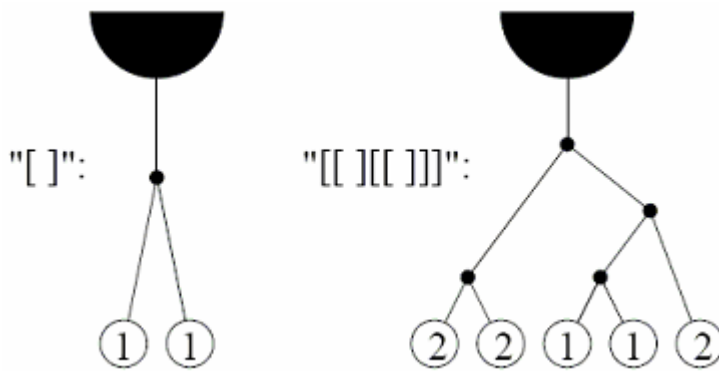
Input	Output
AF AE AB AC AD	5
AB AC AD AE AF	AB 2
	AC 2
	AD 2
	AE 2
	AF 2

BÀI 117. CÂN ĐĨA

Cho một cân đĩa ban đầu chỉ gồm một sợi dây. Mỗi sợi dây lại có thể phân nhánh thành 2 sợi dây con, một sợi dây bị phân nhánh sẽ được kí hiệu là [] (ở trong dấu [] là 2 nhánh của sợi dây đó, nó có thể chứa các dấu [] khác nếu sợi dây bị phân nhánh nhiều lần). Ở các sợi dây mà không bị phân nhánh nữa, người ta sẽ treo vào đó các quả cân.

Các bạn có thể hình dung mô hình cân này sẽ tạo thành một cây nhị phân, ví dụ như trong hình. Biết rằng: cây nhị phân này có độ sâu không quá 25. Nhiệm vụ của bạn là treo các quả cân sao cho cân cân bằng và tổng trọng lượng các quả cân là bé nhất có thể.

Lưu ý: Cân sẽ cân bằng nếu như với mỗi sợi dây bị phân nhánh thì tổng trọng lượng sợi dây bên trái bằng tổng trọng lượng sợi dây bên phải. Ban đầu có một sợi dây nên ít nhất có một quả cân trọng lượng



Dữ liệu vào:

- Dòng đầu tiên chứa 1 số nguyên N ($1 \leq N \leq 1000$) là số bộ test.
- Sau đó là N bộ test, mỗi bộ test trên một dòng chứa một xâu các dấu '[' và ']' kí hiệu cho cân đĩa. $0 \leq \text{độ dài xâu} \leq 150$.

Kết quả:

- Với mỗi bộ test, in ra trên 1 dòng chứa số tự tự bộ test, dấu cách, và kết quả bài toán.

Ví dụ:

Input	Output
3	1 2
[]	2 1
[[] []]	3 8

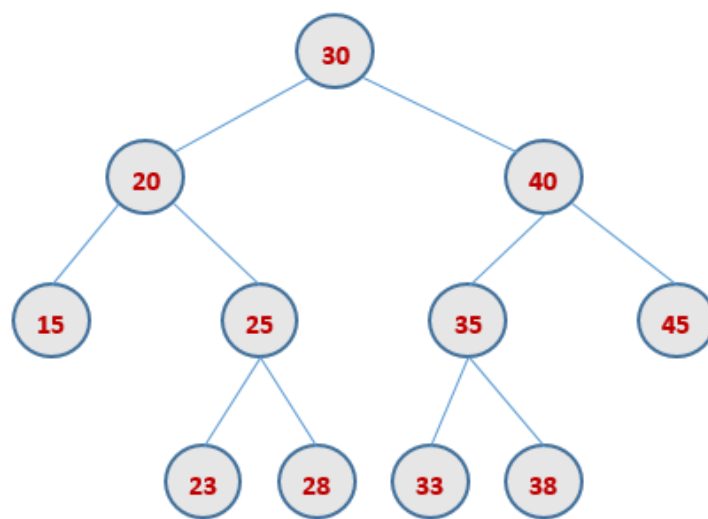
Chú ý: trong input của ví dụ: dòng thứ 2 là một dòng trống

BÀI 118. NODE LÁ

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preoder) của một cây nhị phân tìm kiếm.

Hãy in ra tất cả các node lá của cây ?

Ví dụ với dãy $A[] = \{30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45\}$ là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả: 15, 23, 28, 33, 38, 45.



Input

- Dòng đầu tiên đưa vào số lượng bộ test T ($T \leq 100$).

- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên N ($N \leq 10^4$). Dòng tiếp theo là N số là phép duyệt theo thứ tự trước của cây BST.

Output:

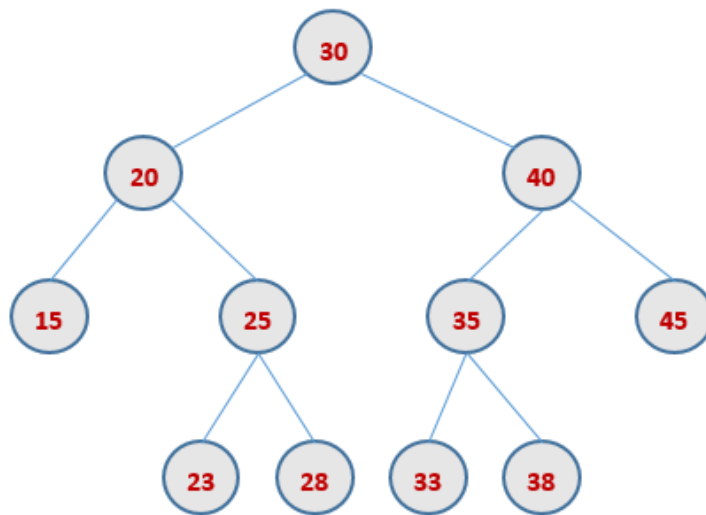
- Đưa ra kết quả mỗi test theo từng dòng sắp xếp theo độ lớn tăng dần.

Input:	Output
2	1 7 50
6	15 23 28 33 38 45
10 5 1 7 40 50	
11	
30 20 15 25 23 28 40 35 33 38 45	

BÀI 119. ĐỘ SÂU CỦA CÂY

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preoder) của một cây nhị phân tìm kiếm. Hãy tìm độ sâu của cây ?

Ví dụ với dãy $A[] = \{30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45\}$ là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả là 3.



Input:

- Dòng đầu tiên đưa vào số lượng bộ test T ($T \leq 100$).
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên N ($N \leq 10^4$). Dòng tiếp theo là N số là phép duyệt theo thứ tự trước của cây BST.

Output:

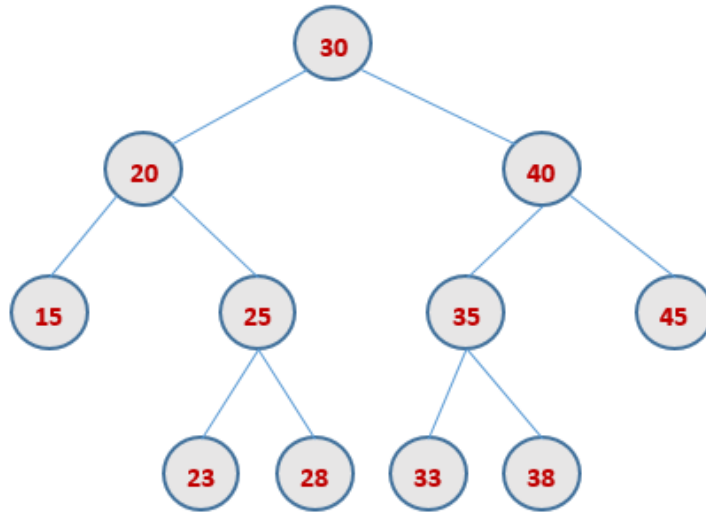
- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	2
6	3
10 5 1 7 40 50	
11	
30 20 15 25 23 28 40 35 33 38 45	

BÀI 120. NODE TRUNG GIAN

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preoder) của một cây nhị phân tìm kiếm. Hãy đưa ra số các node trung gian của cây ?

Ví dụ với dãy $A[] = \{30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45\}$ là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả là 5 bao gồm các node: 30, 20, 25, 40, 35.



Input:

- Dòng đầu tiên đưa vào số lượng bộ test T ($T \leq 100$).
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên N ($N \leq 10^4$). Dòng tiếp theo là N số là phép duyệt theo thứ tự trước của cây BST.

Output:

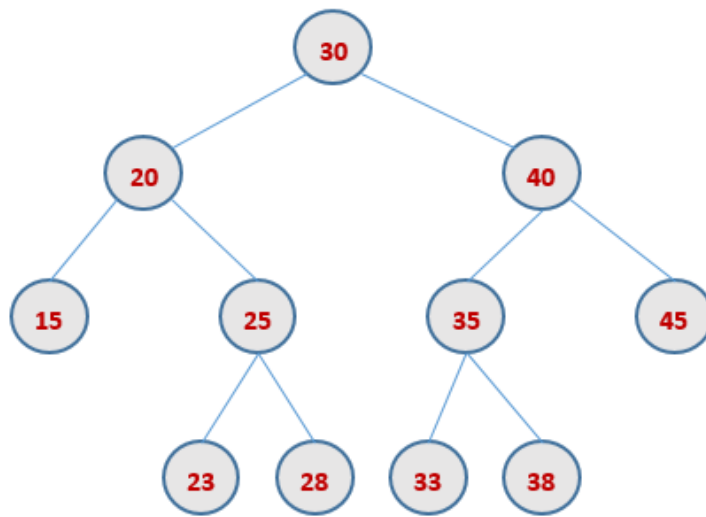
- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	3
6	5
10 5 1 7 40 50	
11	
30 20 15 25 23 28 40 35 33 38 45	

BÀI 121. DUYỆT THEO THỨ TỰ GIỮA

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preoder) của một cây nhị phân tìm kiếm. Hãy in ra phép duyệt theo thứ tự giữa của cây?

Ví dụ với dãy $A[] = \{30, 20, 15, 25, 23, 28, 40, 35, 33, 38, 45\}$ là phép duyệt cây theo thứ tự trước sẽ cho ta kết quả là : 15, 20, 23, 25, 28, 30, 33, 35, 38, 40, 45.



Input:

- Dòng đầu tiên đưa vào số lượng bộ test T ($T \leq 100$).
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất là số tự nhiên N ($N \leq 10^4$). Dòng tiếp theo là N số là phép duyệt theo thứ tự trước của cây BST.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	2
6	1 5 7 10 40 50
10 5 1 7 40 50	15 20 23 25 28 30 33 35 38 40 45
11	
30 20 15 25 23 28 40 35 33 38 45	