

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIÊN THÔNG
KHOA CÔNG NGHỆ THÔNG TIN 1



BÀI TẬP
CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT

THƯ VIỆN PTT

Biên soạn:

ThS. Nguyễn Mạnh Sơn

Hà Nội – 2022

MỤC LỤC

MỤC LỤC	1
LỜI MỞ ĐẦU	6
CHƯƠNG 1. THUẬT TOÁN SINH VÀ QUAY LUI	7
1.1. XÂU NHỊ PHÂN KẾ TIẾP	7
1.2. TẬP CON KẾ TIẾP	7
1.3. HOÁN VỊ KẾ TIẾP	8
1.4. XÂU AB CÓ ĐỘ DÀI N	9
1.5. XÂU NHỊ PHÂN CÓ K BIT 1	9
1.6. PHÂN TÍCH SỐ 1	10
1.7. TẬP CON LIỀN KỀ PHÍA TRƯỚC	11
1.8. XÂU NHỊ PHÂN TRƯỚC	11
1.9. TỔ HỢP TIẾP THEO	12
1.10. SỐ THÚ TỰ HOÁN VỊ	13
1.11. SỐ THÚ TỰ TỔ HỢP	13
1.12. ĐẶT TÊN - 1	14
1.13. ĐẶT TÊN - 2	15
1.14. PHÁT LỘC	16
1.15. HOÁN VỊ DÃY SỐ	17
1.16. LIỆT KÊ TỔ HỢP	17
1.17. HOÁN VỊ XÂU KÝ TỰ	18
1.18. DÃY CON TỔNG BẰNG K	19
1.19. ĐỒI CHỖ CÁC CHỮ SỐ	20
1.20. CHỌN SỐ TỪ MA TRẬN VUÔNG CẤP N	20
1.21. TỔ HỢP SỐ CÓ TỔNG BẰNG X	21
1.22. MÁY ATM	22
1.23. DI CHUYỂN TRONG MA TRẬN	22
1.24. SỐ NGUYÊN TỐ	23
1.25. SẮP XẾP QUÂN HẬU 1	24
1.26. SẮP XẾP QUÂN HẬU 2	24
1.27. BIỂU THỨC TOÁN HỌC	25
1.28. NGƯỜI DU LỊCH	26
CHƯƠNG 2. GIẢI THUẬT THAM LAM	27
2.1. ĐỒI TIỀN	27
2.2. NHÀM CHỮ SỐ	27
2.3. TÌM MAX	28

2.4. TỔNG NHỎ NHẤT	29
2.5. CHIA MẢNG THÀNH HAI MẢNG CON CÓ TỔNG LỚN NHẤT	29
2.6. SẮP XẾP THAM LAM	30
2.7. GIÁ TRỊ NHỎ NHẤT CỦA BIÊU THỨC	31
2.8. SẮP XẾP CÔNG VIỆC 1	32
2.9. SẮP XẾP CÔNG VIỆC 2	33
2.10. NỐI DÂY 1	34
2.11. NỐI DÂY 2	35
2.12. SẮP ĐẶT XÂU KÝ TỰ 1	35
2.13. SẮP ĐẶT XÂU KÝ TỰ 2	36
2.14. SỐ KHỎI LẬP PHƯƠNG	37
2.15. SỐ NHỎ NHẤT	37
2.16. GIÁ TRỊ NHỎ NHẤT CỦA XÂU	38
2.17. SỐ MAY MẮN	39
2.18. PHÂN SỐ ĐƠN VỊ	39
2.19. GIÁ TRỊ LỚN NHẤT	40
CHƯƠNG 3. KỸ THUẬT CHIA VÀ TRỊ.....	41
3.1. LŨY THỪA	41
3.2. LŨY THỪA ĐÀO	41
3.3. ĐÉM DÂY	42
3.4. GẤP ĐÔI DÂY SỐ	42
3.5. DÂY XÂU FIBONACI	43
3.6. ĐÉM SỐ BÍT 1	43
3.7. HỆ CƠ SỐ K	44
3.8. SỐ FIBONACCI THU N	45
3.9. LŨY THỪA MA TRẬN 1	45
3.10. DÂY CON LIÊN TIẾP CÓ TỔNG LỚN NHẤT	46
3.11. TÍCH HAI SỐ NHỊ PHÂN	47
3.12. TÍCH ĐA THỨC	47
3.13. CẶP NGHỊCH THẾ	48
3.14. DÂY XÂU NHỊ PHÂN	49
3.15. KÝ TỰ THỨ K TRONG XÂU	50
3.16. KHOẢNG CÁCH NHỎ HƠN K	50
CHƯƠNG 4. GIẢI THUẬT QUY HOẠCH ĐỘNG	52
4.1. XÂU CON CHUNG DÀI NHẤT	52
4.2. DÂY CON CHUNG DÀI NHẤT CỦA BA XÂU	52
4.3. DÂY CON TĂNG DÀI NHẤT	53
4.4. SỐ BUỚC ÍT NHẤT	53
4.5. TỔNG LỚN NHẤT CỦA DÂY CON TĂNG DÀN	54
4.6. TỔNG LỚN NHẤT CỦA DÂY CON KHÔNG KÈ NHAU	55
4.7. DÂY CON CÓ TỔNG BẰNG S	56
4.8. TẬP CON BẰNG NHAU	56
4.9. DÂY CON DÀI NHẤT CÓ TỔNG CHIA HẾT CHO K	57

4.10. SỐ CÓ TỔNG CHỮ SỐ BẰNG K	58
4.11. TỔ HỢP C(N, K)	58
4.12. BẬC THANG	59
4.13. CATALAN NUMBER	60
4.14. TÍNH P(N,K)	60
4.15. XÂU CON ĐỒI XỨNG DÀI NHẤT	61
4.16. HÌNH VUÔNG LỚN NHẤT	62
4.17. ĐƯỜNG ĐI NHỎ NHẤT	63
4.18. KÝ TỰ GIỐNG NHAU	64
4.19. TỔNG CÁC XÂU CON	64
4.20. TỔNG BẰNG K	65
4.21. CÁI TÚI	66
4.22. BIẾN ĐỒI XÂU	67
4.23. GIẢI MÃ	68
4.24. DI CHUYỂN VỀ GỐC TỌA ĐỘ	68
4.25. TỔNG BÌNH PHƯƠNG	69
CHƯƠNG 5. NGĂN XẾP	70
5.1. KIỂM TRA BIỂU THỨC SỐ HỌC	70
5.2. ĐÉM SỐ DẤU NGOẶC ĐỒI CHIỀU	70
5.3. BIỂU THỨC TƯƠNG ĐƯƠNG	71
5.4. XÓA DẤU NGOẶC	72
5.5. SO SÁNH BIỂU THỨC	73
5.6. BIẾN ĐỒI TRUNG TỐ - HẬU TỐ	74
5.7. BIẾN ĐỒI TIỀN TỐ - TRUNG TỐ	74
5.8. BIẾN ĐỒI TIỀN TỐ - HẬU TỐ	75
5.9. BIẾN ĐỒI HẬU TỐ - TIỀN TỐ	76
5.10. BIẾN ĐỒI HẬU TỐ - TRUNG TỐ	76
5.11. TÍNH GIÁ TRỊ BIỂU THỨC HẬU TỐ	77
5.12. TÍNH GIÁ TRỊ BIỂU THỨC TIỀN TỐ	77
5.13. TÍNH TOÁN GIÁ TRỊ BIỂU THỨC TRUNG TỐ	78
5.14. BIỂU THỨC TĂNG GIẢM	79
5.15. PHẦN TỬ BÊN PHẢI NHỎ HƠN	80
5.16. HÌNH CHỮ NHẬT LỚN NHẤT	81
5.17. DÃY NGOẶC ĐÚNG DÀI NHẤT	82
5.18. PHẦN TỬ CÓ SỐ LẦN XUẤT HIỆN NHIỀU HƠN BÊN PHẢI	82
5.19. NHỊP CHỨNG KHOÁN	83
5.20. GIẢI MÃ XÂU KÝ TỰ	84
5.21. BIỂU THỨC ĐÚNG DÀI NHẤT	85
CHƯƠNG 6. HÀNG ĐỢI	86
6.1. SỐ NHỊ PHÂN TỪ 1 ĐẾN N	86
6.2. SỐ 0 VÀ SỐ 9	86
6.3. SỐ BDN 1	87
6.4. SỐ BDN 2	87

6.5. BIẾN ĐỔI S – T	88
6.6. BIẾN ĐỔI SỐ TỰ NHIÊN	89
6.7. KHOẢNG CÁCH XÂU KÝ TỰ	90
6.8. TÌM SỐ K THỎA MÃN ĐIỀU KIỆN	91
6.9. DI CHUYỂN TRÁNH VẬT CẨN	92
6.10. GIEO MẦM	93
6.11. DI CHUYỂN TRONG KHÔNG GIAN	94
6.12. SỐ LỘC PHÁT 1	96
6.13. SỐ LỘC PHÁT 2	96
6.14. SỐ LỘC PHÁT 3	97
6.15. BIẾN ĐỔI SỐ NGUYÊN TỐ	97
6.16. DI CHUYỂN TRONG MA TRẬN	98
6.17. QUÂN MÃ	99
6.18. QUAY HÌNH VUÔNG	100
CHƯƠNG 7. ĐỒ THỊ	102
7.1. DFS TRÊN ĐỒ THỊ VÔ HƯỚNG	102
7.2. BFS TRÊN ĐỒ THỊ VÔ HƯỚNG	103
7.3. ĐƯỜNG ĐI THEO DFS VỚI ĐỒ THỊ VÔ HƯỚNG	103
7.4. ĐƯỜNG ĐI THEO BFS TRÊN ĐỒ THỊ VÔ HƯỚNG	104
7.5. ĐÊM SỐ THÀNH PHẦN LIÊN THÔNG	105
7.6. TÌM SỐ THÀNH PHẦN LIÊN THÔNG VỚI BFS	105
7.7. KIỂM TRA TÍNH LIÊN THÔNG MẠNH	106
7.8. SỐ LUỢNG HÒN ĐẢO	107
7.9. LIỆT KÊ ĐỈNH TRÙ	108
7.10. LIỆT KÊ CẠNH CẦU	108
7.11. KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ VÔ HƯỚNG	109
7.12. KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG	110
7.13. KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG VỚI DFS	110
7.14. KIỂM TRA ĐỒ THỊ CÓ PHẢI LÀ CÂY HAY KHÔNG	111
7.15. LIỆT KÊ ĐỈNH TRÙ	112
7.16. KIỂM TRA CHU TRÌNH	113
7.17. CHUYỂN DANH SÁCH KÈ SANG MA TRẬN KÈ	113
7.18. CHUYỂN MA TRẬN KÈ SANG DANH SÁCH KÈ	114
7.19. DFS TRÊN ĐỒ THỊ CÓ HƯỚNG	115
7.20. BFS TRÊN ĐỒ THỊ CÓ HƯỚNG	115
7.21. ĐƯỜNG ĐI THEO DFS VỚI ĐỒ THỊ CÓ HƯỚNG	116
7.22. ĐƯỜNG ĐI THEO BFS TRÊN ĐỒ THỊ CÓ HƯỚNG	117
7.23. KIỂM TRA ĐƯỜNG ĐI	117
7.24. ĐƯỜNG ĐI HAMILTON	118
7.25. ĐỒ THỊ HAI PHÍA	119
7.26. KẾT BẠN	120
7.27. MẠNG XÃ HỘI	121
7.28. CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN DFS	123
7.29. CÂY KHUNG CỦA ĐỒ THỊ THEO THUẬT TOÁN BFS	124

7.30. DIJKSTRA	125
7.31. FLOYD	126
7.32. DI CHUYỀN TRÊN BẢNG SỐ	127
7.33. KRUSKAL	128
7.34. NỐI ĐIỂM	129
CHƯƠNG 8. CÂY NHỊ PHÂN	131
8.1. DUYỆT CÂY 1	131
8.2. DUYỆT CÂY THEO MỨC	132
8.3. DUYỆT CÂY 2	133
8.4. DUYỆT CÂY KIỂU XOĂN ỐC	134
8.5. KIỂM TRA NODE LÁ	135
8.6. CÂY NHỊ PHÂN HOÀN HẢO	136
8.7. CÂY NHỊ PHÂN ĐỦ	137
8.8. CÂY NHỊ PHÂN BẰNG NHAU	138
8.9. TỔNG NODE LÁ BÊN PHẢI	139
8.10. TỔNG LỚN NHẤT	140
8.11. BIẾN ĐỔI SANG CÂY NHỊ PHÂN TÌM KIẾM	141
8.12. DUYỆT CÂY NHỊ PHÂN TÌM KIẾM 1	142
8.13. XÂY DỰNG LẠI CÂY NHỊ PHÂN TÌM KIẾM	143
8.14. DUYỆT CÂY NHỊ PHÂN TÌM KIẾM 2	144
8.15. KIỂM TRA CÂY NHỊ PHÂN TÌM KIẾM	144
8.16. NODE LÁ CỦA CÂY NHỊ PHÂN TÌM KIẾM	145
8.17. NODE TRUNG GIAN CỦA CÂY NHỊ PHÂN TÌM KIẾM	146
8.18. ĐỘ SÂU CÂY NHỊ PHÂN TÌM KIẾM	146
8.19. CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG 1	147
8.20. CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG 2	148
8.21. DUYỆT SAU CÂY TÌM KIẾM CÂN BẰNG	149
8.22. Đếm số node lá cây nhị phân tìm kiếm cân bằng	149
TÀI LIỆU THAM KHẢO	151

LỜI MỞ ĐẦU

Cấu trúc dữ liệu và Giải thuật là môn học cơ sở ngành quan trọng trong chương trình giáo dục đại cương ở bậc đại học với sinh viên các ngành Công nghệ thông tin, Điện – Điện tử và Công nghệ Đa phương tiện. Cuốn bài tập này được biên soạn nhằm mục tiêu định hướng quá trình luyện tập của sinh viên theo các mảng kiến thức của môn học.

Tài liệu được cấu trúc thành 8 chương, bao gồm:

Chương 1. Thuật toán Sinh và Quay lui

Chương 2. Giải thuật Tham lam

Chương 3. Kỹ thuật Chia và trị

Chương 4. Giải thuật Quy hoạch động

Chương 5. Ngăn xếp

Chương 6. Hàng đợi

Chương 7. Đồ thị

Chương 8. Cây nhị phân

Bài tập trong mỗi chương được sắp xếp theo mức độ từ dễ đến khó. Nhóm tác giả rất mong sẽ nhận được các ý kiến góp ý của các thầy cô và bạn bè đồng nghiệp để có thể hoàn thiện hơn sách bài tập và góp phần nâng cao chất lượng đào tạo của môn học.

CHƯƠNG 1. THUẬT TOÁN SINH VÀ QUAY LUI

1.1. XÂU NHỊ PHÂN KẾ TIẾP

Cho xâu nhị phân $X[]$, nhiệm vụ của bạn là hãy đưa ra xâu nhị phân tiếp theo của $X[]$. Ví dụ $X[] = "010101"$ thì xâu nhị phân tiếp theo của $X[]$ là “010110”.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu nhị phân X.
- T, X[] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(X) \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input	Output
2	
010101	010110
111111	000000

1.2. TẬP CON KẾ TIẾP

Cho hai số N, K và một tập con K phần tử $X[] = (X_1, X_2, \dots, X_K)$ của 1, 2, .., N. Nhiệm vụ của bạn là hãy đưa ra tập con K phần tử tiếp theo của $X[]$. Ví dụ N=5, K=3, $X[] = \{2, 3, 4\}$ thì tập tiếp theo của $X[]$ là $\{2, 3, 5\}$.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là hai số N và K; dòng tiếp theo đưa vào K phần tử của $X[]$ là một tập con K phần tử của 1, 2, .., N.
- T, K, N, X[] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq K \leq N \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

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

1.3. HOÁN VỊ KẾ TIẾP

Cho số tự nhiên N và một hoán vị $X[]$ của $1, 2, \dots, N$. Nhiệm vụ của bạn là đưa ra hoán vị tiếp theo của $X[]$. Ví dụ $N=5$, $X[] = \{1, 2, 3, 4, 5\}$ thì hoán vị tiếp theo của $X[]$ là $\{1, 2, 3, 5, 4\}$.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là số N; dòng tiếp theo đưa vào hoán vị $X[]$ của $1, 2, \dots, N$.
- T, N, $X[]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

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

1.4. XÂU AB CÓ ĐỘ DÀI N

Xâu ký tự str được gọi là xâu AB nếu mỗi ký tự trong xâu hoặc là ký tự ‘A’ hoặc là ký tự ‘B’. Ví dụ xâu str= ”ABBABB” là xâu AB độ dài 6. Nhiệm vụ của bạn là hãy liệt kê tất cả các xâu AB có độ dài n.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên n.
- T, n thỏa mãn ràng buộc: $1 \leq T \leq 10$; $1 \leq n \leq 10$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng. Mỗi xâu cách nhau 1 khoảng trắng.

Input	Output
2	AA AB BA BB
2	AAA AAB ABA ABB BAA BAB BBA BBB
3	

1.5. XÂU NHỊ PHÂN CÓ K BIT 1

Hãy in ra tất cả các xâu 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 xâu in ra trên một dòng.

Ví dụ:

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

1.6. PHÂN TÍCH SỐ 1

Cho số nguyên dương N. Nhiệm vụ của bạn là hãy liệt kê tất cả các cách phân tích số tự nhiên N thành tổng các số tự nhiên nhỏ hơn hoặc bằng N. Phép hoán vị vừa một cách được xem là giống nhau. Ví dụ với N = 5 ta có kết quả là: (5), (4, 1), (3, 2), (3, 1, 1), (2, 2, 1), (2, 1, 1, 1), (1, 1, 1, 1, 1).

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên N được viết trên một dòng.
- T, n thỏa mãn ràng buộc: $1 \leq T, N \leq 10$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input	Output
2	(4) (3 1) (2 2) (2 1 1) (1 1 1 1)
4	(5) (4 1) (3 2) (3 1 1) (2 2 1) (2 1 1 1) (1 1 1 1 1)
5	

1.7. TẬP CON LIỀN KÈ PHÍA TRƯỚC

Cho hai số N, K và một tập con K phần tử $X[] = (X_1, X_2, \dots, X_K)$ của 1, 2, .., N. Nhiệm vụ của bạn là hãy đưa ra tập con K phần tử trước đó của $X[]$. Ví dụ $N=5$, $K=3$, $X[] = \{2, 3, 5\}$ thì tập con trước đó của $X[]$ là $\{2, 3, 4\}$. Chú ý nếu tập con trong input là đầu tiên thì trước đó là tập con cuối cùng.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là hai số N và K; dòng tiếp theo đưa vào K phần tử của $X[]$ là một tập con K phần tử của 1, 2, .., N.
- $T, K, N, X[]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq K \leq N \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng,

Ví dụ:

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

1.8. XÂU NHỊ PHÂN TRƯỚC

Cho xâu nhị phân $X[]$, nhiệm vụ của bạn là hãy đưa ra xâu nhị phân trước của $X[]$. Ví dụ $X[] = "111111"$ thì xâu nhị phân trước của $X[]$ là “111110”. Với xâu $X[] = "000001"$ thì xâu nhị phân trước của $X[]$ là “000000”. Chú ý: nếu xâu dữ liệu trong input là xâu đầu tiên thì trước nó sẽ là xâu cuối cùng.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu nhị phân X.
- $T, X[]$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq \text{length}(X) \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	010100
010101	111110
111111	

1.9. 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

1.10. SỐ THỨ TỰ HOÁN VỊ

Cho một hoán vị của N số nguyên dương đầu tiên.

Hãy xác định xem đó là hoán vị thứ bao nhiêu nếu liệt kê theo thứ tự tăng dần (tính từ 1).

Input

Dòng đầu ghi số T là số bộ test ($T < 10$)

Mỗi bộ test gồm 2 dòng

- Dòng đầu ghi số nguyên dương N ($3 < N < 10$)
- Dòng tiếp theo ghi một hoán vị của các số nguyên dương từ 1 đến N .

Output

Với mỗi bộ test, ghi ra trên một dòng số thứ tự của hoán vị (tính từ 1, theo thứ tự liệt kê tăng dần).

Ví dụ

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

1.11. SỐ THỨ TỰ TỔ HỢP

Cho một tổ hợp chập K của N số nguyên dương đầu tiên ($2 < K < N < 15$).

Hãy xác định xem đó là tổ hợp thứ bao nhiêu nếu liệt kê tất cả các tổ hợp theo thứ tự tăng dần (tính từ 1).

Input

Dòng đầu ghi số T là số bộ test ($T < 10$)

Mỗi bộ test gồm 2 dòng

- Dòng đầu ghi 2 số nguyên dương N và K ($2 < K < N < 15$)
- Dòng tiếp theo ghi một tổ hợp chập K của các số nguyên dương từ 1 đến N.

Output

Với mỗi bộ test, ghi ra trên một dòng số thứ tự của tổ hợp (tính từ 1, theo thứ tự liệt kê tăng dần).

Ví dụ

Input	Output
2	9
6 4	12
1 3 5 6	.
6 4	
2 3 4 6	

1.12. ĐẶT TÊN - 1

Kỳ thi ICPC có K đội của PTIT tham gia và đội tuyển đang rất đau đầu không biết chọn các cái tên như thế nào cho các đội. Yêu cầu phải đảm bảo tên không có khoảng trống và không được trùng nhau. Sau khi thảo luận, có N cái tên được đề xuất (có thể bị trùng nhau). Với $K < 15$ và $4 < N < 30$.

Hãy liệt kê tất cả danh sách các tổ hợp K cái tên khác nhau có thể được tạo ra theo thứ tự từ điển.

Input

Dòng đầu ghi 2 số N và K.

Tiếp theo là 1 dòng ghi N cái tên, mỗi cái tên có độ dài không quá 15 và cách nhau một khoảng trống. Tất cả đều là ký tự in hoa.

Output

Ghi ra tất cả các tổ hợp tên có thể được lựa chọn theo thứ tự từ điển.

Tức là các tên trong mỗi tổ hợp liệt kê theo thứ tự từ điển và các tổ hợp cũng được liệt kê theo thứ tự từ điển.

Ví dụ

Input	Output
6 2 DONG TAY NAM BAC TAY BAC	BAC DONG BAC NAM BAC TAY DONG NAM DONG TAY NAM TAY

1.13. ĐẶT TÊN - 2

Vương quốc PTIT sử dụng bảng chữ cái gồm N chữ cái Latinh viết hoa. Quy tắc đặt tên của gia đình Hoàng gia PTIT là chọn ra K chữ cái (không trùng nhau) và sắp xếp lại theo thứ tự từ điển.

Hãy liệt kê tất cả các cái tên có thể có của gia đình Hoàng gia PTIT

Input

- Dòng đầu ghi số bộ test T (không quá 10)
- Mỗi bộ test ghi 2 số N và K ($3 < K < N < 16$)

Output

- Với mỗi bộ test, ghi ra tất cả các cái tên có thể được tạo ra, mỗi kết quả viết trên một dòng.

Ví dụ

Input	Output
1	AB
4 2	AC
	AD
	BC
	BD
	CD

1.14. PHÁT LỘC

Một xâu ký tự số chỉ bao gồm các chữ số 6 và 8 sẽ được gọi là “phát lộc” nếu thỏa mãn các điều kiện sau:

- Độ dài xâu ít nhất là 6
- Chữ số đầu tiên là chữ số 8, chữ số cuối cùng là chữ số 6
- Không có 2 chữ số 8 nào ở cạnh nhau
- Không có nhiều hơn 3 chữ số 6 ở cạnh nhau.

Viết chương trình liệt kê các xâu ký tự phát lộc độ dài N theo thứ tự tăng dần.

Input

Chỉ có 1 dòng ghi số N ($5 < N < 16$).

Output

Ghi ra các xâu ký tự phát lộc độ dài N, mỗi xâu trên một dòng.

Ví dụ

Input	Output
6	866686 866866 868666

1.15. HOÁN VỊ DÃY SỐ

Cho dãy số A[] có N phần tử là các số nguyên dương khác nhau từng đôi một. Hãy liệt kê tất cả các hoán vị của dãy số A[] theo thứ tự tăng dần, tức là hoán vị đầu tiên có giá trị tăng dần từ trái qua phải, hoán vị cuối cùng giảm dần từ trái qua phải.

Input

Dòng đầu ghi số N ($1 < N < 9$)

Dòng thứ 2 ghi N số của dãy A[] ($0 < A[i] < 10000$)

Output

Ghi mỗi hoán vị của dãy số trên một dòng

Ví dụ

Input	Output
3	77 88 99
88 77 99	77 99 88
	88 77 99
	88 99 77
	99 77 88
	99 88 77

1.16. LIỆT KÊ TỔ HỢP

Cho dãy số A[] có N phần tử. Hãy liệt kê tất cả các tổ hợp chập K của tập các phần tử khác nhau trong A[]. Các tổ hợp cần liệt kê theo thứ tự từ điển (tức là trong mỗi tổ hợp thì giá trị từ nhỏ đến lớn, và tổ hợp sau lớn hơn tổ hợp trước).

Input

Dòng đầu ghi hai số N và K.

Dòng thứ 2 ghi N số của mảng A[]. Các giá trị không quá 1000.

Dữ liệu đảm bảo số phần tử khác nhau của A[] không quá 20 và K không quá 10.

Output

Ghi ra lần lượt các tổ hợp tìm được, mỗi tổ hợp trên một dòng.

Ví dụ

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

1.17. HOÁN VỊ XÂU KÝ TỰ

Cho xâu ký tự S bao gồm các ký tự in hoa khác nhau. Hãy đưa ra tất cả các hoán vị của xâu ký tự S. Ví dụ S="ABC" ta có kết quả {ABC ACB BAC BCA CAB CBA}.

Input:

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

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input	Output
2	AB BA
AB	
ABC	ABC ACB BAC BCA CAB CBA

1.18. DÃY CON TỔNG BẰNG K

Cho dãy số $A[] = (a_1, a_2, \dots, a_n)$ và số tự nhiên K . Hãy đưa ra tất cả các dãy con của dãy số $A[]$ sao cho tổng các phần tử của dãy con đó đúng bằng K . Các phần tử của dãy số $A[]$ được giả thuyết là nguyên dương và không có các phần tử giống nhau. Ví dụ với dãy con $A[] = \{5, 10, 15, 20, 25\}$, $K = 50$ ta có 3 dãy con $\{5, 10, 15, 20\}$, $\{5, 20, 25\}$, $\{10, 15, 25\}$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số N là số lượng phần tử của dãy số $A[]$ và số K ; dòng tiếp theo đưa vào N phần tử của dãy số $A[]$.
- $T, N, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10$; $1 \leq K, A[i] \leq 100$.

Output:

- Đưa ra tất cả các dãy con của dãy số $A[]$ thỏa mãn yêu cầu bài toán theo thứ tự từ điển, trong đó mỗi dãy con được bao bởi các ký tự [,]. Nếu không có dãy con nào thỏa mãn yêu cầu bài toán, hãy đưa ra -1.

Input	Output
2	
5 50	[5 10 15 20] [5 20 25] [10 15 25]
5 10 15 20 25	[8 9 14 22] [8 14 15 16] [15 16 22]
8 53	
15 22 14 26 32 9 16 8	

1.19. ĐỔI CHỖ CÁC CHỮ SỐ

Cho số tự nhiên K và xâu ký tự các chữ số S. Nhiệm vụ của bạn là đưa ra số lớn nhất bằng cách thực hiện nhiều nhất K lần đổi chỗ các ký tự trong S. Ví dụ K =3 và S = “1234567” ta được “7654321”.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là số K; dòng tiếp theo là xâu ký tự S.
- T, K, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq K \leq 10$; $1 \leq \text{length}(S) \leq 7$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input	Output
3	
4	
1234567	7654321
3	
3435335	5543333
2	
1034	4301

1.20. CHỌN SỐ TỪ MA TRẬN VUÔNG CẤP N

Cho ma trận vuông $C_{i,j}$ cấp N ($1 \leq i, j \leq N < 10$) 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 4 3	2
1 3 6	1 3 2
4 2 4	3 2 1

1.21. TỔ HỢP SỐ CÓ TỔNG BẰNG X

Cho mảng $A[]$ gồm N số nguyên dương phân biệt và số X . Nhiệm vụ của bạn là tìm phép tổ hợp các số trong mảng $A[]$ có tổng bằng X . Các số trong mảng $A[]$ có thể được sử dụng nhiều lần. Mỗi tổ hợp các số của mảng $A[]$ được in ra theo thứ tự không giảm các số. Ví dụ với $A[] = \{2, 4, 6, 8\}$, $X = 8$ ta có các tổ hợp các số như sau:

[2, 2, 2, 2], [2, 2, 4], [2, 6], [4, 4], [8].

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số N và X ; dòng tiếp theo đưa vào N số của mảng $A[]$; các số được viết cách nhau một vài khoảng trắng.
- $T, N, X, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 10$; $1 \leq X, A[i] \leq 100$. $N \leq 20$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng. Mỗi đường tổ hợp được bao bởi cặp ký tự []. Đưa ra -1 nếu không có tổ hợp nào thỏa mãn yêu cầu bài toán.

Input	Output
1 4 8	[2 2 2] [2 2 4] [2 6] [4 4] [8]

2 4 6 8	
---------	--

1.22. 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, mỗi tờ tiền chỉ được dùng một lần).

Input: Dòng đầu tiên ghi số bộ test T ($T < 10$). Mỗi bộ test 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: Với mỗi bộ test ghi ra số tờ tiền ít nhất phải trả.

Nếu không thể tìm được kết quả, in ra -1.

Ví dụ

Input	Output
1	
3 5	
1 4 5	1

1.23. DI CHUYỂN TRONG MA TRẬN

Cho ma trận $A[M][N]$. Nhiệm vụ của bạn là đếm tất cả các đường đi từ phần tử $A[0][0]$ đến phần tử $A[M-1][N-1]$. Bạn chỉ được phép dịch chuyển xuống dưới hoặc sang phải phần tử liền kề với vị trí hiện tại.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số M, N tương ứng với số hàng và số cột của ma trận; dòng tiếp theo đưa

vào các phần tử của ma trận $A[][]$; các số được viết cách nhau một vài khoảng trống.

- $T, M, N, A[i][j]$ thỏa mãn ràng buộc: $1 \leq T \leq 10; 1 \leq M, N, A[i][j] \leq 100$.

Output:

- Đưa ra số cách di chuyển của mỗi test theo từng dòng.
- Giải thích test 1: Có 3 cách di chuyển là [1 4 5 6], [1 2 5 6] và [1 2 3 6].

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

1.24. SỐ NGUYÊN TỐ

Cho ba số N, P, S . Trong đó, P là một số nguyên tố. Nhiệm vụ của bạn là đưa ra tất cả N số nguyên tố lớn hơn P có tổng bằng S . Ví dụ với $S = 28, P=7, N=2$ ta có kết quả $11 + 17 = 28$. Với $N = 3, P = 2, S = 23$ ta có kết quả : $\{3, 7, 13\}, \{5, 7, 11\}$

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ ba số S, P, N được viết trên một dòng.
- S, P, N thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq N \leq 10; 2 \leq S, P \leq 200$.

Output:

- Với mỗi test, dòng đầu tiên in ra số lượng đáp án tìm được. Mỗi dòng tiếp theo in ra kết quả tìm được theo thứ tự từ điển.

Input	Output
	1
	11 17
2	2
2 7 28	3 7 13
3 2 23	5 7 11

1.25. 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. Ván đài 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: Dòng đầu ghi số bộ test T ($T < 5$). Mỗi bộ test ghi một số nguyên dương n duy nhất (không quá 10)

Output: Ghi kết quả mỗi bộ test trên một dòng. Số cách đặt quân hậu.

Ví dụ:

Input	Output
1	
4	2

1.26. 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 48 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64	260

~~1.27. BIỂU THỨC TOÁN HỌC~~

Cho 5 số nguyên dương A, B, C, D, E. Bạn có thể hoán vị các phần tử cho nhau, hãy đặt các dấu biểu thức +, -, * sao cho biểu thức sau đúng:

$$[[A o(1) B] o(2) C] o(3) D] o(4) E = 23$$

Trong đó: o(1) ... o(4) là các phép toán +, -, *.

Input:

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

Mỗi test gồm 5 số nguyên dương A, B, C, D, E có giá trị không vượt quá 100.

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

Ví dụ:

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

3	NO
1 1 1 1 1	YES
1 2 3 4 5	YES
2 3 5 7 11	

1.28. 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 $C[1...n, 1...n]$ 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

CHƯƠNG 2. GIẢI THUẬT THAM LAM

2.1. ĐỔI TIỀN

Tại ngân hàng có các mệnh giá bằng 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000. Tổng số tiền cần đổi có giá trị bằng N. Hãy xác định xem 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 ($1 \leq N \leq 100000$).

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

Ví dụ:

Input:	Output
2	1
70	7
121	12

2.2. 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 không bao giờ tính toán sai, nhưng thỉnh thoảng cậu ta chép các con số một cách không 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	Input: 1430 4862	Input: 16796 58786
Ouput: 36 37	Ouput: 6282 6292	Ouput: 74580 85582

2.3. TÌM MAX

Cho mảng A[] gồm N phần tử. Nhiệm vụ của bạn là

$$\max = \sum_{i=0}^{n-1} A_i * i$$

tìm

bằng cách sắp đặt lại các phần tử trong mảng.

Chú ý, kết quả của bài toán có thể rất lớn vì vậy bạn hãy đưa ra kết quả lấy modulo với $10^9 + 7$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng N; dòng tiếp theo đưa vào N số A[i] tương ứng với các phần tử của mảng A[]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N, A[i] \leq 10^7$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	40
5	8
5 3 2 4 1	

3	
1 2 3	

2.4. TỔNG NHỎ NHẤT

Cho mảng A[] gồm các số từ 0 đến 9. Nhiệm vụ của bạn là tìm tổng nhỏ nhất của hai số được tạo bởi các số trong mảng A[]. Chú ý, tất cả các số trong mảng A[] đều được sử dụng để tạo nên hai số.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng N; dòng tiếp theo đưa vào N số A[i] tương ứng với các phần tử của mảng A[]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 20$; $0 \leq A[i] \leq 9$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2 6 6 8 4 5 2 3 5 5 3 0 7 4	604 82

2.5. CHIA MẢNG THÀNH HAI MẢNG CON CÓ TỔNG LỚN NHẤT

Cho mảng A[] gồm N số nguyên không âm và số K. Nhiệm vụ của bạn là hãy chia mảng A[] thành hai mảng con có kích cỡ K và N-K sao cho hiệu giữa tổng hai mảng con là lớn nhất. Ví dụ với mảng A[] = {8, 4, 5, 2, 10}, K=2 ta có kết quả là 17 vì mảng A[] được

chia thành hai mảng $\{4, 2\}$ và $\{8, 5, 10\}$ có hiệu của hai mảng con là $23 - 6 = 17$ là lớn nhất.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng N và số K; dòng tiếp theo đưa vào N số $A[i]$ tương ứng với các phần tử của mảng $A[]$; các số được viết cách nhau một vài khoảng trắng.
- $T, N, K, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq K < N \leq 50$; $0 \leq A[i] \leq 1000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
5 2	
8 4 5 2 10	
8 3	
1 1 1 1 1 1 1 1	

2.6. SẮP XẾP THAM LAM

Cho mảng $A[]$ gồm N số và thực hiện các thao tác theo nguyên tắc dưới đây:

- Ta chọn một mảng con sao cho phần tử ở giữa của mảng con cũng là phần tử ở giữa của mảng $A[]$ (trong trường hợp N lẻ).
- Đảo ngược mảng con đã chọn trong mảng $A[]$. Ta được phép chọn mảng con và phép đảo ngược mảng con bao nhiêu lần tùy ý.

Ví dụ với mảng $A[] = \{1, 6, 3, 4, 5, 2, 7\}$ ta có câu trả lời là Yes vì: ta chọn mảng con $\{3, 4, 5\}$ và đảo ngược để nhận được mảng $A[] = \{1, 6, 5, 4, 3, 2, 7\}$, chọn tiếp mảng con

$\{6, 5, 4, 3, 2\}$ và đảo ngược ta nhận được mảng $A[] = \{1, 2, 3, 4, 5, 6, 7\}$. Hãy cho biết ta có thể sắp xếp được mảng $A[]$ bằng cách thực hiện các thao tác kê trên hay không?

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào số phần tử của mảng N ; dòng tiếp theo đưa vào N số $A[i]$ tương ứng với các phần tử của mảng $A[]$; các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 50$; $0 \leq A[i] \leq 1000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2 7 1 6 3 4 5 2 7	Yes
7 1 6 3 4 5 7 2	No

2.7. GIÁ TRỊ NHỎ NHẤT CỦA BIỂU THỨC

Cho mảng $A[], B[]$ đều có N phần tử. Nhiệm vụ của bạn là tìm giá trị nhỏ nhất của biểu thức $P = A[0]*B[0] + A[1]*B[1] + .. + A[N-1]*B[N-1]$ bằng cách tráo đổi vị trí các phần tử của cả mảng $A[]$ và $B[]$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào số phần tử của mảng N ; dòng tiếp theo đưa vào N số $A[i]$; dòng cuối cùng đưa vào N số $B[i]$ các số được viết cách nhau một vài khoảng trống.
- $T, N, A[i], B[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^7$; $0 \leq A[i], B[i] \leq 10^{18}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
7	
1 6 3 4 5 2 7	45
1 1 1 2 3 4 3	27
7	
1 6 3 5 5 2 2	
0 1 9 0 1 2 3	

2.8. SẮP XẾP CÔNG VIỆC 1

Cho hệ gồm N hành động. Mỗi hành động được biểu diễn như một bộ đôi $\langle S_i, F_i \rangle$ tương ứng với thời gian bắt đầu và thời gian kết thúc của mỗi hành động. Hãy tìm phương án thực hiện nhiều nhất các hành động được thực hiện bởi một máy hoặc một người sao cho hệ không xảy ra mâu thuẫn.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào số lượng hành động N; dòng tiếp theo đưa vào N số S_i tương ứng với thời gian bắt đầu mỗi hành động; dòng cuối cùng đưa vào N số F_i tương ứng với thời gian kết thúc mỗi hành động; các số được viết cách nhau một vài khoảng trắng.
- T, N, S_i , F_i thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N, F_i, S_i \leq 1000$.

Output:

- Đưa số lượng lớn nhất các hành động có thể được thực thi bởi một máy hoặc một người.

Ví dụ:

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

2.9. SẮP XẾP CÔNG VIỆC 2

Cho N công việc. Mỗi công việc được biểu diễn như một bộ 3 số nguyên dương $\langle \text{JobId}, \text{Deadline}, \text{Profit} \rangle$, trong đó JobId là mã của việc, Deadline là thời gian kết thúc của việc, Profit là lợi nhuận đem lại nếu hoàn thành việc đó đúng hoặc trước thời gian. Thời gian tối thiểu để hoàn thành mỗi công việc là **1 đơn vị thời gian**. Hãy cho biết lợi nhuận lớn nhất có thể thực hiện các việc với giả thiết mỗi việc được thực hiện đơn lẻ.

Input:

- Đòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là số lượng Job N; phần thứ hai đưa vào $3 \times N$ số tương ứng với N job.
- T, N, JobId, Deadline, Profit thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 1000$; $1 \leq \text{JobId} \leq 1000$; $1 \leq \text{Deadline} \leq 1000$; $1 \leq \text{Profit} \leq 1000$.

Output:

- Đưa số lượng công việc tương ứng và lợi nhuận lớn nhất có thể đạt được.

Ví dụ:

Input	Output
2	
4	
1 4 20	2 60
2 1 10	2 127
3 1 40	

4	1	30
5		
1	2	100
2	1	19
3	2	27
4	1	25
5	1	15

2.10. NỐI DÂY 1

Cho N sợi dây với độ dài khác nhau được lưu trong mảng $A[]$. Nhiệm vụ của bạn là nối N sợi dây thành một sợi sao cho tổng chi phí nối dây là nhỏ nhất. Biết chi phí nối sợi dây thứ i và sợi dây thứ j là tổng độ dài hai sợi dây $A[i]$ và $A[j]$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào số lượng sợi dây N ; dòng tiếp theo đưa vào N số $A[i]$ là độ dài của các sợi dây; các số được viết cách nhau một vài khoảng trắng.
- $T, N, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^6$; $0 \leq A[i] \leq 10^6$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
4	
4 3 2 6	29
5	62
4 2 7 6 9	

2.11. NỐI DÂY 2

Cho N sợi dây với độ dài khác nhau được lưu trong mảng $A[]$. Nhiệm vụ của bạn là nối N sợi dây thành một sợi sao cho tổng chi phí nối dây là nhỏ nhất. Biết chi phí nối sợi dây thứ i và sợi dây thứ j là tổng độ dài hai sợi dây $A[i]$ và $A[j]$.

Dữ liệu vào

Dòng đầu ghi số bộ test T ($T < 10$). Mỗi bộ test gồm 2 dòng. Dòng đầu tiên là số nguyên N ($N \leq 2 * 10^6$).

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

Kết quả

In ra đáp án của bộ test trên từng dòng, theo modulo $10^9 + 7$.

Ví dụ:

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

2.12. SẮP ĐẶT XÂU KÝ TỰ 1

Cho xâu ký tự S bao gồm các ký tự in thường. Nhiệm vụ của bạn là kiểm tra xem ta có thể sắp đặt lại các ký tự trong S để hai ký tự giống nhau đều không kề nhau hay không? Đưa ra 1 nếu có thể sắp đặt lại các ký tự trong S thỏa mãn yêu cầu bài toán, ngược lại đưa ra -1.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu ký tự S được viết trên một dòng.
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(S) \leq 10000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	
geeksforgeeks	1
bbbabaacd	1
bbbbbb	-1

2.13. SẮP ĐẶT XÂU KÝ TỰ 2

Cho xâu ký tự S bao gồm các ký tự và số D. Nhiệm vụ của bạn là kiểm tra xem ta có thể sắp đặt lại các ký tự trong S để tất cả các ký tự giống nhau đều có khoảng cách là D hay không? Đưa ra 1 nếu có thể sắp đặt lại các ký tự trong S thỏa mãn yêu cầu bài toán, ngược lại đưa ra -1.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là số D; dòng tiếp theo là xâu S.
- T, S, D thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(S) \leq 10000$; $1 \leq D \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
2	
ABB	1
2	-1
AAA	

2.14. SỐ KHỐI LẬP PHƯƠNG

Một số X được gọi là số khối lập phương nếu X là lũy thừa bậc 3 của số Y ($X = Y^3$). Cho số nguyên dương N, nhiệm vụ của bạn là tìm số khối lập phương lớn nhất bằng cách loại bỏ đi các chữ số của N. Ví dụ số 4125 ta có kết quả là $125 = 5^3$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số tự nhiên N được viết trên một dòng.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^{18}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng. Nếu không tìm được đáp án in ra -1.

Ví dụ:

Input	Output
2	125
4125	-1
976	

2.15. SỐ NHỎ NHẤT

Cho hai số nguyên dương S và D, trong đó S là tổng các chữ số và D là số các chữ số của một số. Nhiệm vụ của bạn là tìm số nhỏ nhất thỏa mãn S và D? Ví dụ với S = 9, D = 2 ta có số nhỏ nhất thỏa mãn S và D là 18.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ 2 số S và D được viết trên một dòng.
- T, S, D thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq S, D \leq 1000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng. Nếu không có đáp án, in ra -1.

Ví dụ:

Input	Output
2	18
9 2	299
20 3	

2.16. GIÁ TRỊ NHỎ NHẤT CỦA XÂU

Cho xâu ký tự S. Ta gọi giá trị của xâu S là tổng bình phương số lần xuất hiện mỗi ký tự trong S. Hãy tìm giá trị nhỏ nhất của xâu S sau khi thực hiện K lần loại bỏ ký tự.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là số K; phần thứ hai là một xâu ký tự S được viết trên một dòng.
- T, S, K thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(S) \leq 10000$; $1 \leq K \leq 1000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	6
2	2
ABCCBC	
2	
AAAB	

2.17. 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	47
11	-1
10	

2.18. PHÂN SỐ ĐƠN VỊ

Một phân số đơn vị nếu tử số của phân số đó là 1. Mọi phân số nguyên dương đều có thể biểu diễn thành tổng các phân số đơn vị. Ví dụ $\frac{2}{3} = \frac{1}{2} + \frac{1}{6}$. Cho phân số nguyên dương P/Q bất kỳ ($P < Q$), hãy biểu diễn phân số nguyên dương thành tổng phân số đơn vị với số hạng tử là ít nhất.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ đôi tử số P và mẫu số Q của phân số nguyên dương được viết trên một dòng.
- T, P, Q thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq P, Q \leq 100$.

Output:

- Đưa ra đáp án tìm được trên 1 dòng, theo dạng “ $1/a + 1/b + \dots$ ”

Ví dụ:

Input	Output
2	$1/2 + 1/6$
2 3	
1 3	$1/3$

2.19. GIÁ TRỊ LỚN NHẤT

Cho dãy số nguyên $A[]$ có N phần tử.

- Gọi $f(i,j) = |a_i| + |a_{i+1}| + \dots + |a_j|$
- Gọi $g(i,j) = a_i + a_{i+1} + \dots + a_j$

Với tất cả các cặp $1 \leq i \leq j \leq N$.

Hãy tính **giá trị lớn nhất** của $f(i,j) + g(i,j)$.

Input

Dòng đầu ghi số N ($1 \leq N \leq 50000$)

Dòng thứ 2 ghi N số nguyên của dãy $A[]$

Output

Ghi ra giá trị lớn nhất của $f(i,j) + g(i,j)$

Ví dụ

Input	Output
5 -3 5 -10 8 -2	26

CHƯƠNG 3. KỸ THUẬT CHIA VÀ TRỊ

3.1. 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	16
4 2	

3.2. LŨY THỪA ĐẢO

Cho mảng số N . Ta gọi số đảo của N là R . Hãy tìm lũy thừa R của N . Đưa ra kết quả của bài toán dưới dạng modulo với $10^9 + 7$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm là số N được ghi trên một dòng.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^{10}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	4
2	864354781
12	

3.3. ĐẾM DÃY

Cho số nguyên dương n . Hãy cho biết có bao nhiêu dãy số nguyên dương có tổng các phần tử trong dãy bằng n .

Dữ liệu vào: dòng đầu tiên chứa số nguyên T là số bộ dữ liệu, mỗi bộ dữ liệu ghi một số nguyên dương n duy nhất không quá 10^{18} .

Kết quả: Mỗi bộ dữ liệu ghi ra một số nguyên duy nhất là số dư của kết quả tìm được khi chia cho **123456789**.

Ví dụ:

Input	Output
1	
3	4

3.4. 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 $[1] - [1 2 1] - [1 2 1 3 1 2 1]$.

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.

Ví dụ:

Input	Output
2	2
3 2	4
4 8	

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].

3.5. 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	A
6 4	B
8 19	

3.6. ĐẾM SỐ BÍT 1

Cho số nguyên dương N. Mỗi bước, bạn sẽ biến đổi N thành $[N/2]$, $N \bmod 2$, $[N/2]$. 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] à [3, 1, 3] à [1, 1, 1, 1, 3] à [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, 1, 1, 0, 1].

3.7. HỆ CƠ SỐ K

Cho hai số A, B ở hệ cơ số K. Hãy tính tổng hai số đó ở hệ cơ số K.

Input: Dòng đầu ghi số bộ test T ($T < 10$). Mỗi bộ test ghi 3 số K,A,B.

($2 \leq K \leq 10$; A và B nếu biểu diễn trong hệ cơ số 10 đều nhỏ hơn 10^9)

Output: In ra tổng của A và B trong hệ cơ số K

Ví dụ:

Input	Output
1	11
2 1 10	

3.8. SỐ FIBONACCI THỨ N

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

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

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

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 bao 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	
2	
6	
20	

3.9. LŨY THƯA MA TRẬN 1

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 bao 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	

3.10. DÃY CON LIÊN TIẾP CÓ TỔNG LỚN NHẤT

Cho mảng A[] gồm N số có cả các số âm và số dương. Nhiệm vụ của bạn là tìm mảng con liên tục có tổng lớn nhất của mảng. Ví dụ với mảng A[]={-2, -5, 6, -2, -3, 1, 5, -6} ta có kết quả là 7 tương ứng với dãy con {6, -2, -3, 1, 5}.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng thứ nhất đưa vào hai số N tương ứng với số phần tử của mảng; dòng tiếp theo đưa vào N số A[i]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 100$; $-100 \leq A[i] \leq 100$.

Output:

- Đưa ra tổng con liên tục lớn nhất của mỗi test theo từng dòng.

Ví dụ:

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

3.11. TÍCH HAI SỐ NHỊ PHÂN

Cho hai xâu nhị phân biểu diễn hai số. Nhiệm vụ của bạn là đưa ra tích của hai số. Ví dụ với xâu $S1 = "1100"$ và $S2 = "1010"$ ta sẽ có kết quả là 120.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 2 hai xâu nhị phân $S1$, $S2$ được viết trên một dòng.
- $T, S1, S2$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(S1), \text{length}(S2) \leq 30$.

Output:

- Đưa ra tích của mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
1100 01	
01 01	12
	1

3.12. TÍCH ĐA THÚC

Cho hai đa thức P và Q được biểu diễn như một mảng bao gồm các hệ số của đa thức. Ví dụ với $P(x) = 5 + 0x^1 + 10x^2 + 6x^3$ được biểu diễn như mảng $P[] = \{5, 0, 10, 6\}$. Hãy đưa ra đa thức $R = P \times Q$ theo các hệ số của R với cách biểu diễn như trên.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng thứ nhất đưa vào hai số M, N tương ứng với lũy thừa lớn nhất của đa thức P và Q; dòng tiếp theo đưa vào M số là hệ số của đa thức P; dòng cuối cùng đưa vào M số là hệ số của đa thức Q.
- $T, M, N, P[i], Q[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq M, N \leq 100$; $1 \leq P[i], Q[i] \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	2 0 10 4 12 8
4 3	4 36 14 39 79 23 34 35
1 0 3 2	
2 0 4	
5 4	
1 9 3 4 7	
4 0 2 5	

3.13. CẶP NGHỊCH THẾ

Cho mảng $A[]$ gồm N phần. Ta gọi cặp nghịch thế của mảng $A[]$ là số các cặp i, j sao cho $i < j$ và $A[i] > A[j]$. Đối với mảng đã được sắp xếp thì số cặp nghịch thế bằng 0. Mảng đã sắp theo thứ tự giảm dần có số đảo ngược cực đại. Nhiệm vụ của bạn là hãy đưa ra số cặp nghịch thế của mảng $A[]$ gồm N phần tử.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.

- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số N tương ứng với số phần tử của mảng A[]; phần thứ 2 là N số của mảng A[]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^7$; $1 \leq A[i] \leq 10^{18}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
5	
2 4 1 3 5	3
5	10
5 4 3 2 1	

3.14. DÃY XÂU NHỊ PHÂN

Người ta tạo ra một dãy xâu ký tự nhị phân $X[]$ trong đó:

$$X[1] = "0"$$

$$X[2] = "1"$$

$$X[i] = X[i-2] + X[i-1] \text{ với } i > 2$$

Với phép cộng (+) là phép nối hai xâu với nhau.

Cho hai số tự nhiên N và K ($1 < N < 93$; K đảm bảo trong phạm vi của xâu $X[N]$).

Hãy xác định ký tự thứ K trong xâu $X[N]$ là ký tự ‘0’ hay ký tự ‘1’.

Input: Dòng 1 ghi số bộ test. Mỗi bộ test ghi trên một dòng 2 số nguyên N và K.

Output: Ghi ra màn hình kết quả tương ứng với từng bộ test.

Ví dụ:

Input	Output
2	0

3 1
4 3

1

3.15. KÝ TỰ THỨ K TRONG XÂU

Xâu ký tự S được tạo ra bằng cách bổ sung dần các ký tự chữ cái Tiếng Anh in hoa như sau.

- Bước 1: Chỉ có chữ cái A
- Bước 2: Thêm chữ cái B vào giữa 2 chữ A => S = "ABA"
- Bước 3: Thêm chữ cái C vào giữa 2 xâu đã có ở bước 2: S = "ABACABA"

Cứ như vậy cho đến bước thứ N ($0 < N < 26$)

Hãy xác định ký tự thứ K trong bước biến đổi thứ N là chữ cái gì?

Input:

- Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).
- Mỗi test gồm số nguyên dương N và K ($1 \leq N \leq 25$, $1 \leq K \leq 2^N - 1$).

Output:

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

Ví dụ:

Input:	Output
2	B
3 2	D
4 8	

3.16. KHOẢNG CÁCH NHỎ HƠN K

Cho mảng A[] gồm n số nguyên dương và số K. Hãy đếm số các cặp phần tử có hiệu nhỏ hơn K. Ví dụ A[] = {1, 10, 4, 2}, K=3 ta nhận được kết quả là 2 tương ứng với các cặp (1, 2), (4, 2).

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Mỗi bộ test gồm hai dòng: dòng đầu tiên ghi số n và số K; dòng tiếp theo là n số A[i] ; các số được viết cách nhau một vài khoảng trắng.
- T, n, k, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^4$; $1 \leq k \leq 10^3$; $1 \leq A[i] \leq 10^5$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
4 3	
1 10 4 2	2
3 5	3
2 3 4	

CHƯƠNG 4. GIẢI THUẬT QUY HOẠCH ĐỘNG

4.1. XÂU CON CHUNG DÀI NHẤT

Cho 2 xâu S1 và S2. Hãy tìm xâu con chung dài nhất của 2 xâu này (*các phần tử không nhất thiết phải liên tiếp nhau*).

Input: Dòng đầu tiên là số lượng bộ test T ($T \leq 20$). Mỗi test gồm hai dòng, mô tả xâu S1 và S2, mỗi xâu có độ dài không quá 1000 và chỉ gồm các chữ cái in hoa.

Output: Với mỗi test, in ra độ dài dãy con chung dài nhất trên một dòng.

Ví dụ:

Input	Output
2	
AGGTAB	
GXTXAYB	
AA	
BB	

Giải thích test 1: Dãy con chung là G, T, A, B.

4.2. DÃY CON CHUNG DÀI NHẤT CỦA BA XÂU

Cho ba xâu ký tự X, Y, Z. Nhiệm vụ của bạn là tìm độ dài dãy con chung dài nhất có mặt trong cả ba xâu.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào độ dài xâu X, Y, X; dòng tiếp theo đưa vào ba xâu X, Y, Z.
- T, X, Y, Z thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{size}(X), \text{size}(Y), \text{size}(Z) \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
5 8 13	5
geeks geeksfor geeksforgeeks	3
7 6 5	
abcd1e2 bc12ea bd1ea	

4.3. 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 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 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 $A[1], A[2], \dots, A[N]$ ($1 \leq A[i] \leq 1000$).

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

Ví dụ:

Input	Output
6	4
1 2 5 4 6 2	

4.4. SỐ BUỚC ÍT NHẤT

Cho mảng $A[]$ gồm N số nguyên. Nhiệm vụ của bạn là sắp xếp lại mảng số với số lượng bước là ít nhất. Tại mỗi bước, bạn chỉ được phép chèn phần tử bất kỳ của mảng vào vị trí bất kỳ trong mảng. Ví dụ $A[] = \{2, 3, 5, 1, 4, 7, 6\}$ sẽ cho ta số phép chèn ít nhất là 3

bằng cách lấy số 1 chèn trước số 2, lấy số 4 chèn trước số 5, lấy số 6 chèn trước số 7 ta nhận được mảng được sắp.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là một số N; dòng tiếp theo đưa vào N số của mảng A[]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 1000$; $1 \leq A[i] \leq 1000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

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

4.5. TỔNG LỚN NHẤT CỦA DÃY CON TĂNG DÀN

Cho dãy số A[] gồm N số. Nhiệm vụ của bạn là tìm tổng lớn nhất của dãy con được sắp theo thứ tự tăng dần của dãy A[]. Ví dụ với dãy A[] = {1, 101, 2, 3, 100, 4, 5} ta có kết quả là $106 = 1 + 2 + 3 + 100$. Với dãy A[] = {10, 7, 5} ta có kết quả là 10. Với dãy A[] = {1, 2, 3, 5} ta có kết quả là 11.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào N là số phần tử của dãy A[]; dòng tiếp theo đưa vào N số A[i]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $0 \leq A[i] \leq 10^3$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	
7	
1 101 2 3 100 4 5	106
3	10
10 7 5	11
4	
1 2 3 5	

4.6. TỔNG LỚN NHẤT CỦA DÃY CON KHÔNG KÈ NHAU

Cho dãy số $A[]$ gồm N phần tử. Hãy tìm tổng lớn nhất của dãy con của dãy số $A[]$ sao cho dãy con không có hai số cạnh nhau trong $A[]$. Ví dụ với $A[] = \{6, 7, 1, 3, 8, 2, 4\}$ ta có kết quả là 19 tương ứng với tổng của dãy con $\{6, 1, 6, 4\}$ không có hai phần tử nào kề nhau trong $A[]$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là một số N ; dòng tiếp theo đưa vào N số $A[i]$; các số được viết cách nhau một vài khoảng trắng.
- $T, N, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^6$; $1 \leq A[i] \leq 10^7$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output

2	
6	
5 5 10 100 10 5	110
4	13
3 2 7 10	

4.7. 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 ghi số bộ test T ($T < 10$). Mỗi bộ test có hai dòng, dòng đầu tiên ghi hai số nguyên dương N và S ($0 < N \leq 200$) và S ($0 < S \leq 40000$). 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: Với mỗi bộ test, nếu bài toán vô nghiệm thì in ra “NO”, ngược lại in ra “YES”

Ví dụ:

Input	Output
2 5 6 1 2 4 3 5 10 15 2 2 2 2 2 2 2 2 2	YES NO

4.8. TẬP CON BẰNG NHAU

Cho tập các số $A[] = (a_1, a_2, \dots, a_n)$. Hãy kiểm tra xem ta có thể chia tập $A[]$ thành hai tập con sao cho tổng các phần tử của hai tập con bằng nhau hay không. Đưa ra YES nếu có thể thực hiện được, ngược lại đưa ra NO.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .

- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào số N là số lượng phần tử của dãy số A[]; dòng tiếp theo đưa vào N phần tử của dãy số A[].
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 100$; $1 \leq A[i] \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input	Output
2	
4	
1 5 11 5	YES
3	NO
1 3 5	

4.9. 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 ghi số bộ test T ($T < 10$). Mỗi bộ test gồm 2 dòng. Dòng đầu tiên chứa hai số n, k. Dòng tiếp theo ghi n số của dãy A. Các số đều không vượt quá 100.

Output: Gồm 1 dòng duy nhất ghi số lượng phần tử của dãy con dài nhất thỏa mãn. Dữ liệu vào luôn đảm bảo sẽ có ít nhất một dãy con có tổng chia hết cho k.

Ví dụ:

Input	Output
1	
10 3	9
2 3 5 7 9 6 12 7 11 15	

4.10. 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
2 5	5
3 6	21

Giải thích test 1: 11 và 20.

Giải thích test 2: 14, 23, 32, 41.

4.11. 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	
5 2	10
10 3	120

4.12. BẬC THANG

Một chiếc cầu thang có N bậc. Mỗi bước, bạn được phép bước lên trên tối đa K bước. 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
4 2	5

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).

4.13. CATALAN NUMBER

Catalan Number là dãy số thỏa mãn biểu thức:

$$C_n = \begin{cases} 0 & \text{nếu } n = 0 \\ \sum_{i=0}^{n-1} C_i C_{n-i-1} & \text{nếu } n > 0 \end{cases}$$

Dưới đây là một số số Catalan với $n=0, 1, 2, \dots : 1, 1, 2, 5, 14, 42, 132, 429, \dots$ Cho số tự nhiên N. Nhiệm vụ của bạn là đưa ra số Catalan thứ N.

Input:

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

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	
5	
4	
10	

4.14. TÍNH P(N,K)

$P(n, k)$ là số phép biều diễn các tập con có thứ tự gồm k phần tử của tập gồm n phần tử. Số $P(n, k)$ được định nghĩa theo công thức sau:

$$P(n, k) = \begin{cases} 0 & \text{nếu } k > n \\ \frac{n!}{(n-k)!} = n \cdot (n-1) \dots (n-k+1) & \text{nếu } k \leq n \end{cases}$$

Cho số hai số n, k. Hãy tìm P(n,k) theo modulo $10^9 + 7$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một cặp số n, k được viết trên một dòng.
- T, n, k thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n, k \leq 1000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
5 2	20
4 2	12

4.15. XÂU CON ĐỐI XUNG DÀI NHẤT

Cho xâu S chỉ bao gồm các ký tự viết thường và dài không quá 1000 ký tự.

Hãy tìm xâu con đối xứng dài nhất của S.

Input:

- Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).
- Mỗi test gồm một xâu S có độ dài không vượt quá 1000, chỉ gồm các ký tự thường.

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

Ví dụ:

Input	Output
2	
abcbadd	5
aaaaa	5

4.16. 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.

Ví dụ:

Input:	Output
2 6 5 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	3 0

0 0 0 0 0	
2 2	
0 0	
0 0	

4.17. ĐƯỜNG ĐI NHỎ NHẤT

Cho bảng $A[]$ kích thước $N \times M$ (N hàng, M cột). Bạn được phép đi xuống dưới, 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 3 3 1 2 3 4 8 2 1 5 3	8

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

4.18. KÝ TỰ GIỐNG NHAU

Giả sử bạn cần viết N ký tự giống nhau lên màn hình. Bạn chỉ được phép thực hiện ba thao tác dưới đây với chi phí thời gian khác nhau:

- Thao tác insert: chèn một ký tự với thời gian là X .
- Thao tác delete: loại bỏ ký tự cuối cùng với thời gian là Y .
- Thao tác copying: copy và paste tất cả các ký tự đã viết để số ký tự được nhân đôi với thời gian là Z .

Hãy tìm thời gian ít nhất để có thể đưa ra màn hình N ký tự giống nhau. Ví dụ với $N = 9$, $X = 1$, $Y = 2$, $Z = 1$ ta có kết quả là 5 bằng cách thực hiện: insert, insert, copying, copying, insert.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào N là số các ký tự giống nhau cần viết lên màn hình; dòng tiếp theo đưa vào bộ ba số X , Y , Z tương ứng với thời gian thực hiện ba thao tác; các số được viết cách nhau một vài khoảng trắng.
- T , N , X , Y , Z thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 100$; $1 \leq X, Y, Z \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2 9 1 2 1 10 2 5 4	5 14

4.19. TỔNG CÁC XÂU CON

Cho số nguyên dương N được biểu diễn như một xâu ký tự số. Nhiệm vụ của bạn là tìm tổng của tất cả các số tạo bởi các xâu con của N. Ví dụ N="1234" ta có kết quả là 1670 = $1 + 2 + 3 + 4 + 12 + 23 + 34 + 123 + 234 + 1234$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một số N.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^{12}$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
1234	1670
421	491

4.20. TỔNG BẰNG K

Cho một mảng A[] gồm N số nguyên và số K. Tính số cách lấy tổng các phần tử của A[] để bằng K. Phép lấy lặp các phần tử hoặc sắp đặt lại các phần tử được chấp thuận. Ví dụ với mảng A[] = {1, 5, 6}, K = 7 ta có 6 cách sau:

$$7 = 1 + 1 + 1 + 1 + 1 + 1 + 1 \text{ (lặp số 1 7 lần)}$$

$$7 = 1 + 1 + 5 \text{ (lặp số 1)}$$

$$7 = 1 + 5 + 1 \text{ (lặp và sắp đặt lại số 1)}$$

$$7 = 1 + 6$$

$$7 = 6 + 1$$

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.

- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào N và K; dòng tiếp theo đưa vào N số của mảng A[]; các số được viết cách nhau một vài khoảng trắng.
- T, N, K, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 1000$; $1 \leq A[i] \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng. Khi kết quả quá lớn đưa ra kết quả dưới dạng modulo với $10^9 + 7$.

Ví dụ:

Input	Output
2	
3 7	
1 5 6	
4 14	
12 3 1 9	6 150

4.21. CÁI TÚI

Một người có cái túi thể tích V ($V < 1000$). Anh ta có N đồ vật cần mang theo ($N \leq 1000$), mỗi đồ vật có thể tích là $A[i]$ ($A[i] \leq 100$) và giá trị là $C[i]$ ($C[i] \leq 100$). Hãy xác định tổng giá trị lớn nhất của các đồ vật mà người đó có thể mang theo, sao cho tổng thể tích không vượt quá V.

Input

- Dòng đầu ghi số bộ test T ($T < 10$)
- Mỗi bộ test gồm ba dòng. Dòng đầu ghi 2 số N và V. Dòng tiếp theo ghi N số của mảng A. Sau đó là một dòng ghi N số của mảng C.
- Dữ liệu vào luôn đảm bảo không có đồ vật nào có thể tích lớn hơn V.

Output

- Với mỗi bộ test, ghi trên một dòng giá trị lớn nhất có thể đạt được.

Ví dụ

Input	Output
1 15 10 5 2 1 3 5 2 5 8 9 6 3 1 4 7 8 1 2 3 5 1 2 5 8 7 4 1 2 3 2 1	15

4.22. BIẾN ĐỔI XÂU

Cho hai xâu ký tự str1, str2 bao gồm các ký tự in thường và các thao tác dưới đây:

- **Insert:** chèn một ký tự bất kỳ vào str1.
- **Delete:** loại bỏ một ký tự bất kỳ trong str1.
- **Replace:** thay một ký tự bất kỳ trong str1.

Nhiệm vụ của bạn là đếm số các phép Insert, Delete, Replace ít nhất thực hiện trên str1 để trở thành str2.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ đôi hai xâu str1 và str2.
- T, str1, str2 thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length(str1)}, \text{length(str2)} \leq 100$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
1 geek gesek	1

4.23. GIẢI MÃ

Một bản tin M đã mã hóa bí mật thành các con số theo ánh xạ như sau: ‘A’->1, ‘B’->2, ..., ‘Z’->26. Hãy cho biết có bao nhiêu cách khác nhau để giải mã bản tin M. Ví dụ với bản mã M=“123” nó có thể được giải mã thành ABC (1 2 3), LC (12 3), AW(1 23).

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là một xâu ký tự số M.
- T, M thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(M) \leq 40$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
123	3 2
2563	

4.24. DI CHUYỂN VỀ GỐC TỌA ĐỘ

Giả sử bạn đang ở điểm có tọa độ nguyên dương (n, m) và cần dịch chuyển về tọa độ $(0,0)$. Mỗi bước dịch chuyển bạn chỉ được phép dịch chuyển đến tọa độ $(n-1, m)$ hoặc $(n, m-1)$; Từ ô $(0, m)$, hoặc $(n, 0)$ thì có thể di chuyển 1 bước để về gốc $(0,0)$.

Hãy đếm số cách bạn có thể dịch chuyển về tọa độ $(0,0)$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test là bộ n, m được viết cách nhau một vài khoảng trắng.
- T, n, m thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n, m \leq 25$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	10
3 2	84
3 6	1
3 0	

4.25. TỔNG BÌNH PHƯƠNG

Mọi số nguyên dương N đều có thể phân tích thành tổng các bình phương của các số nhỏ hơn N. Ví dụ số $100 = 10^2$ hoặc $100 = 5^2 + 5^2 + 5^2 + 5^2$. Cho số nguyên dương N. Nhiệm vụ của bạn là tìm số lượng ít nhất các số nhỏ hơn N mà có tổng bình phương bằng N.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi test là một số tự nhiên N được viết trên 1 dòng.
- T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	1
100	3
6	1
25	

CHƯƠNG 5. NGĂN XẾP

5.1. KIỂM TRA BIỂU THỨC SỐ HỌC

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length}(\text{exp}) \leq 20$.

Ví dụ:

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

5.2. ĐẾM SỐ DẤU NGOẶC ĐỔI CHIỀU

Cho một xâu chỉ gồm các kí tự ‘(,)’ và có độ dài chẵn. Hãy đếm số lượng dấu ngoặc cần phải đổi chiều ít nhất, sao cho xâu 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 xâu 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
)()())	

5.3. BIỂU THỨC TƯƠNG ĐƯƠNG

Cho biểu thức đúng P chỉ bao gồm các phép toán +, -, các toán hạng cùng với các ký tự ‘(’, ‘)’. Hãy bỏ tất cả các ký tự ‘(’, ‘)’ trong P để nhận được biểu thức tương đương. Ví dụ với $P = a - (b + c)$ ta có kết quả $P = a - b - c$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức P được viết trên một dòng.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, P thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(P) \leq 10^3$.

Ví dụ:

Input	Output
2 a-(b+c) a-(b-c-(d+e))-f	a-b-c a-b+c+d+e-f

5.4. 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

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

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

Output: In ra tất cả các biểu thức khác nhau thỏa mãn điều 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
--	---------

5.5. SO SÁNH BIỂU THỨC

Cho P1, P2 là hai biểu thức đúng chỉ bao gồm các ký tự mở ngoặc ‘(’ hoặc đóng ngoặc ‘)’ và các toán hạng in thường. Nhiệm vụ của bạn là định xem P1 và P2 có giống nhau hay không.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào P1, dòng tiếp theo đưa vào P2.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, P thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(P) \leq 100$.

Ví dụ:

Input	Output
2 -(a+b+c) -a-b-c a-b-(c-d) a-b-c-d	YES NO

5.6. BIẾN ĐỔI TRUNG TỐ - HẬU TỐ

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length}(\text{exp}) \leq 10$.
- Các phép toán bao gồm +, -, *, /, ^. Phép lũy thừa có ưu tiên cao hơn nhân chia và cộng trừ.

Ví dụ:

Input	Output
2	
(A+(B+C))	ABC++
((A*B)+C)	AB*C+

5.7. BIẾN ĐỔI TIỀN TỐ - TRUNG TỐ

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 10^6$.

Ví dụ:

Input	Output
2 *+AB-CD *-A/BC-/AKL	$((A+B)*(C-D))$ $((A-(B/C))*(A/K)-L)$

5.8. BIẾN ĐỔI TIỀN TỐ - HẬU TỐ

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 10^6$.

Ví dụ:

Input	Output
2 *+AB-CD *-A/BC-/AKL	AB+CD-* ABC/-AK/L-*

5.9. BIẾN ĐỔI HẬU TỐ - TIỀN TỐ

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 10^6$.

Ví dụ:

Input	Output
2	
AB+CD-*	*+AB-CD
ABC/-AK/L-*	*-A/BC-/AKL

5.10. BIẾN ĐỔI HẬU TỐ - TRUNG TỐ

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp.
- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 10^6$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 10^6$.

Ví dụ:

Input	Output
2	
ABC++	(A+(B+C))
AB*C+	((A*B)+C)

5.11. TÍNH GIÁ TRỊ BIỂU THỨC HẬU TỐ

Hãy viết chương trình chuyển tính toán giá trị của biểu thức hậu tố.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức hậu tố exp. Các số xuất hiện trong biểu thức là các số đơn có 1 chữ số.

Output:

- Đưa ra kết quả mỗi test theo từng dòng, chỉ lấy giá trị phần nguyên.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 20$.

Ví dụ:

Input	Output
2	
231*+9-	-4
875*+9-	34

5.12. TÍNH GIÁ TRỊ BIỂU THỨC TIỀN TỐ

Hãy viết chương trình tính toán giá trị của biểu thức tiền tố.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức tiền tố exp. Các số xuất hiện trong biểu thức là các số đơn có 1 chữ số.

Output:

- Đưa ra kết quả mỗi test theo từng dòng, chỉ lấy giá trị phần nguyên.

Ràng buộc:

- T, exp thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq \text{length(exp)} \leq 20$.

Ví dụ:

Input	Output
2	8
$-+8/632$	25
$-+7*45+20$	

5.13. TÍNH TOÁN GIÁ TRỊ BIỂU THỨC TRUNG TỐ

Cho biểu thức trung tố 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	

5.14. BIỂU THỨC TĂNG GIẢM

Cho dãy ký tự S chỉ bao gồm các ký tự I hoặc D. Ký tự I được hiểu là tăng (Increasing) ký tự D được hiểu là giảm (Decreasing). Sử dụng các số từ 1 đến 9, hãy đưa ra số nhỏ nhất được đoán nhận từ S. Chú ý, các số không được phép lặp lại. Dưới đây là một số ví dụ mẫu:

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

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test là một xâu S
- T, S thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(S) \leq 8$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input:	Output:
4	12
I	21
D	321
	321654798

DD	
DDIDDDIID	

5.15. PHẦN TỬ BÊN PHẢI NHỎ HƠN

Cho mảng A[] gồm n phần tử. Hãy đưa ra các phần tử nhỏ hơn tiếp theo của phần tử lớn hơn đầu tiên phần tử hiện tại. Nếu phần tử hiện tại không có phần tử lớn hơn tiếp theo ta xem là -1. Nếu phần tử không có phần tử nhỏ hơn tiếp theo ta cũng xem là -1. Ví dụ với mảng A[] = {5, 1, 9, 2, 5, 1, 7} ta có kết quả là ans = {2, 2, -1, 1, -1, -1, -1} vì:

Next Greater

5	-> 9
1	-> 9
9	-> -1
2	-> 5
5	-> 7
1	-> 7
7	-> -1

Right Smaller

9	-> 2
9	-> 2
-1	-> -1
5	-> 1
7	-> -1
7	-> -1
7	-> -1

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào n là số phần tử của mảng A[], dòng tiếp theo đưa vào n số A[i].

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, n, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n, A[i] \leq 10^6$.

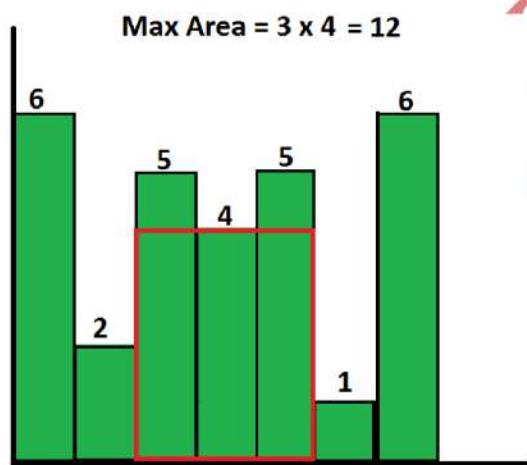
Ví dụ:

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

2	
7	
5 1 9 2 5 1 7	2 2 1 1 -1 -1 -1
8	2 5 5 5 -1 3 -1 -1
4 8 2 1 9 5 6 3	

5.16. 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				

5.17. 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

5.18. PHẦN TỬ CÓ SỐ LẦN XUẤT HIỆN NHIỀU HƠN BÊN PHẢI

Cho mảng A[] gồm n phần tử. Nhiệm vụ của bạn là tìm phần tử gần nhất phía bên phải có số lần xuất hiện lớn hơn phần tử hiện tại. Nếu không có phần tử nào có số lần xuất hiện lớn hơn phần tử hiện tại hãy đưa ra -1.

Ví dụ với dãy A[] = {1, 1, 2, 3, 4, 2, 1}, ta nhận được kết quả ans[] = {-1, -1, 1, 2, 2, 1, -1} vì số lần xuất hiện mỗi phần tử trong mảng là F = {3, 3, 2, 1, 1, 2, 3} vì vậy phần tử A[0] = 1 có số lần xuất hiện là 3 và không có phần tử nào xuất hiện nhiều hơn 3 nên ans[0] = -1, tương tự như vậy với A[2]=2 tồn tại A[6] = 1 có số lần xuất hiện là 3 nên ans[2] = 1.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test gồm hai dòng, dòng đầu tiên đưa vào số n là số các phần tử của mảng A[]; dòng tiếp theo đưa vào n số của mảng A[]; các phần tử được viết cách nhau một vài khoảng trắng.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, n, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $0 \leq n$, $A[i] \leq 10^6$.

Ví dụ:

Input	Output
1 7 1 1 2 3 4 2 1	-1 -1 1 2 2 1 -1

5.19. NHỊP CHỨNG KHOÁN

Bạn là một nhà đầu tư chứng khoán nổi tiếng. Nhiệm vụ hàng ngày của bạn là tính nhịp tăng giảm của phiên chứng khoán trong N ngày để có thể bắt kịp thị trường. Nhịp chứng khoán của ngày thứ i được định nghĩa là số ngày liên tiếp từ ngày thứ i trở về mà giá chứng khoán bé hơn hoặc bằng với giá chứng khoán của ngày i.

Input: Dòng đầu ghi số bộ test (không quá 10). Mỗi test có 2 dòng.

- Dòng đầu tiên gồm 1 số nguyên N ($1 \leq N \leq 10^5$) là số ngày.
- Dòng tiếp theo gồm N số nguyên A_1, A_2, \dots, A_N ($1 \leq A_i \leq 10^6$) là giá chứng khoán của các ngày.

Output

- In ra N số B_1, B_2, \dots, B_N trong đó B_i là nhịp chứng khoán của ngày thứ i .

Ví dụ:

Input	Output
1 7 100 80 60 70 60 75 85	1 1 1 2 1 4 6

5.20. GIẢI MÃ XÂU KÝ TỰ

Cho xâu ký tự mã hóa str. Hãy viết chương trình giải mã xâu ký tự str. Xâu ký tự mã hóa được thực hiện theo số lần lặp các xâu con của str như sau:

Xâu đầu vào: "abbbbababbbbababbbbab "

Xâu mã hóa : "3[a3[b]1[ab]]"

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T ;
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một xâu mã hóa str được viết trên một dòng.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ràng buộc:

- T, str thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq \text{length}(str) \leq 100$.

Ví dụ:

Input	Output
2 1[b] 3[b2[ca]]	b bcacabcacabcaca

5.21. BIỂU THỨC ĐÚNG DÀI NHẤT

Cho biểu thức P chỉ bao gồm các ký tự mở ngoặc ‘(’ hoặc đóng ngoặc ‘)’. Biểu thức P có thể viết đúng hoặc không đúng. Nhiệm vụ của bạn là tìm **tổng độ dài lớn nhất** của các biểu thức con viết đúng trong P (các biểu thức đúng không nhất thiết phải liên tiếp nhau).

Chú ý: Độ dài của biểu thức đúng ngắn nhất là 2.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T (không quá 100)
- Những dòng tiếp theo mỗi dòng đưa vào một bộ test. Mỗi bộ test là một biểu thức P được viết trên một dòng (độ dài của P không quá 100).

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
4	2
((()	4
00((10
(((00)))	6
0((0)(

CHƯƠNG 6. HÀNG ĐỢI

6.1. SỐ NHỊ PHÂN TỪ 1 ĐẾN N

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 ($n \leq 10000$).

Output:

- Đưa ra kết quả mỗi test trên một dòng.

Ví dụ:

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

6.2. SỐ 0 VÀ SỐ 9

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

Input:

- Dòng đầu tiên ghi lại số lượng test T ($T \leq 100$).
- Những dòng kế tiếp mỗi dòng ghi lại một test. Mỗi test là một số tự nhiên N được ghi trên một dòng ($N \leq 100$).

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	

6.3. SỐ BDN 1

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 nhỏ hơn N. Ví dụ N=100 ta có 4 số BDN bao gồm các số: 1, 10, 11, 100.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một số tự nhiên N.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	2
10	4
100	7
200	

6.4. SỐ BDN 2

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.

Số BDN của N là số P = M*N sao cho P là số BDN. Cho số tự nhiên N ($N < 1000$), hãy tìm số BDN nhỏ nhất của N.

Ví dụ. Với $N=2$, ta tìm được số BDN của N là $P = 5*2=10$. $N = 17$ ta tìm được số BDN của 17 là $P = 653*17=11101$.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một số tự nhiên N.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	10
2	11100
12	11101
17	

6.5. BIẾN ĐỔI S – T

Cho hai số nguyên dương S và T ($S, T < 10000$) và hai thao tác (a), (b) dưới đây:

Thao tác (a): Trừ S đi 1 ($S = S-1$);

Thao tác (b): Nhân S với 2 ($S = S*2$);

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): $2*2 = 4$;

Thao tác (b): $4-1 = 3$;

Thao tác (a): $3*2 = 6$;

Thao tác (b): $6-1 = 5$;

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một bộ đôi S và T.

Output: Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

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

6.6. BIẾN ĐỔI SỐ TỰ NHIÊN

Cho số tự nhiên N ($N < 10^9$) và hai phép biến đổi (a), (b) dưới đây.

- **Thao tác (a):** Trừ N đi 1 ($N=N-1$). Ví dụ $N=17$, thao tác (a) biến đổi $N = N-1 = 16$.
- **Thao tác (b):** $N = \max(u, v)$ nếu $u*v = N$ ($u>1, v>1$). 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 cho 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): $N = N-1 = 17-1 = 16$

Thao tác (b): $16 = \max(4, 4) = 4$

Thao tác (b): $4 = \max(2,2) = 2$

Thao tác (a): $2 = 2-1 = 1$

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng Test;
- T dòng kế tiếp mỗi dòng ghi lại một bộ Test. Mỗi test là một số N.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	4
17	5
50	5
100	

6.7. KHOẢNG CÁCH XÂU KÝ TỰ

Cho tập n xâu ký tự S và hai xâu s, t ∈ S. Ta giả thiết các xâu ký tự S[i] ∈ S có độ dài bằng nhau. Hãy tìm khoảng cách đường đi ngắn nhất từ s đến t. Biết từ một xâu ký tự bất kỳ ta chỉ được phép dịch chuyển đến xâu khác với nó duy nhất 1 ký tự. Ví dụ ta có tập các từ S = { POON, TOON, PLEE, SAME, POIE, PLEA, PLIE, POIN }, s = TOON, t = PLEA ta có độ dài đường đi ngắn nhất là 7 tương ứng với các phép dịch chuyển : TOON → POON → POIN → POIE → PLIE → PLEE → PLEA.

Input:

- Dòng đầu tiên đưa vào số lượng test T ($T \leq 100$).
- Mỗi test được tổ chức thành 2 dòng. Dòng thứ nhất ghi lại n là số từ trong S và hai từ s, t. Dòng thứ 2 đưa vào n xâu xâu ký tự của S; các xâu ký tự được viết cách nhau một vài khoảng trắng, có độ dài không vượt quá 10 kí tự.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
1 8 TOON PLEA POON TOON PLEE SAME POIE PLEA PLIE POIN	7

6.8. TÌM SỐ K THỎA MÃN ĐIỀU KIỆN

Cho hai số nguyên dương L, R. Hãy đưa ra số các số K trong khoảng $[L, R]$ thỏa mãn điều kiện:

- Tất cả các chữ số của K đều khác nhau.
- Tất cả các chữ số của K đều nhỏ hơn hoặc bằng 5.

Ví dụ với $L = 4$, $R = 13$ ta có 5 số thỏa mãn yêu cầu là 4, 5, 10, 12, 13,

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test được là một cặp L, R được viết trên một dòng.
- T, L, R thỏa mãn ràng buộc: $1 \leq T \leq 100$; $0 \leq L \leq R \leq 10^5$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	5
4 13	100

6.9. DI CHUYỂN TRÁNH VẬT CẨN

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?

Input:

- Dòng đầu ghi số bộ test (không quá 10). Mỗi test gồm:
 - Dòng đầu tiên là số nguyên dương N ($1 \leq N \leq 100$).
 - N dòng tiếp theo, mỗi dòng gồm N ký 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 có vật cản.

Output:

- Với mỗi test, in ra một số nguyên là đáp số của bài toán.

Ví dụ:

Input	Output
1	
3	
.X.	
.X.	
...	
	3

0 0 0 2	
---------	--

6.10. GIEO MẦM

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?

Input:

- Dòng đầu ghi số bộ test (không quá 10). Mỗi bộ test gồm:
 - 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.

Output:

- Với mỗi test 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:

Input	Output
2	2
3 5	-1

2 1 0 2 1
1 0 1 2 1
1 0 0 2 1
3 5
2 1 0 2 1
0 0 1 2 1
1 0 0 2 1

6.11. DI CHUYỂN TRONG KHÔNG GIAN

Cho một hình hộp chữ nhật có kích thước $A \times B \times C$, trong đó A là chiều cao, B là chiều rộng và C là chiều dài. Mỗi ô có thể là một ô trống ‘.’ hoặc vật cản ‘#’.

Mỗi bước, bạn được phép di chuyển sang một ô kè bên cạnh (không được đi chéo). Nhiệm vụ của bạn là tìm đường đi ngắn nhất bắt đầu ‘S’ tới vị trí kết thúc ‘E’.

Input:

- Dòng đầu tiên là số lượng bộ test T ($1 \leq N \leq 50$).
- Mỗi test bắt đầu bởi 3 số nguyên A, B, C ($A, B, C \leq 30$).
- Tiếp theo là A khối, mỗi khối gồm B x C kí tự mô tả một lát cắt của hình hộp chữ nhật. Giữa 2 khối có một dấu xuống dòng.

Output:

- In ra một số nguyên là đường đi ngắn nhất từ S tới E. Nếu không di chuyển được, in ra -1.

Ví dụ:

Input	Output
2	11
3 4 5	-1

S....
.###.
.##..
###.#

##.##
##...

#.###
###E
1 3 3
S##
#E#
###

THƯ VIỆN PTIT

6.12. SỐ LỘC PHÁT 1

Một số được gọi là lộc phát nếu chỉ có 2 chữ số 6 và 8. Cho số tự nhiên N. Hãy liệt kê các số lộc phát có không quá N chữ số.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng bộ test ($T < 10$);
- T dòng kế tiếp mỗi dòng ghi số N ($1 < N < 15$).

Output:

- In ra đáp án theo thứ tự giảm dần.

Ví dụ:

Input	Output
2	88 86 68 66 8 6
2	888 886 868 866 688 686 668 666 88 86 68 66 8 6
3	

6.13. SỐ LỘC PHÁT 2

Một số được gọi là lộc phát nếu chỉ có 2 chữ số 6 và 8. Cho số tự nhiên N. Hãy liệt kê các số lộc phát có không quá N chữ số.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng bộ test ($T < 10$);
- T dòng kế tiếp mỗi dòng ghi số N ($1 < N < 15$).

Output:

- Dòng đầu tiên là số lượng số lộc phát tìm được. Dòng thứ hai in đáp án **theo thứ tự tăng dần**.

Ví dụ:

Input	Output
2	6

2	6 8 66 68 86 88
3	14
6 8 66 68 86 88 666 668 686 688 866 868 886 888	

6.14. SỐ LỘC PHÁT 3

Một số được gọi là lộc phát nếu chỉ có 2 chữ số 6 và 8. Cho số tự nhiên N. Hãy liệt kê các số lộc phát có không quá N chữ số.

Input:

- Dòng đầu tiên ghi lại số tự nhiên T là số lượng bộ test ($T < 10$);
- T dòng kê tiếp mỗi dòng ghi số N ($1 < N < 15$).

Output:

- Dòng đầu tiên là số lượng số lộc phát tìm được. Dòng thứ hai in ra đáp án **theo thứ tự giảm dần**.

Ví dụ:

Input	Output
2	6
2	88 86 68 66 8 6
3	14
	888 886 868 866 688 686 668 666 88 86 68 66 8 6

6.15. BIẾN ĐỔI SỐ NGUYÊN TỐ

Cho cặp số S và T là các số nguyên tố có 4 chữ số (Ví dụ S = 1033, T = 8197 là các số nguyên tố có 4 chữ số). Hãy viết chương trình tìm cách dịch chuyển S thành T thỏa mãn đồng thời những điều kiện dưới đây:

- Mỗi phép dịch chuyển chỉ được phép thay đổi một chữ số của số ở bước trước đó (ví dụ nếu S=1033 thì phép dịch chuyển S thành 1733 là hợp lệ);

- b. Số nhận được cũng là một số nguyên tố có 4 chữ số (ví dụ nếu $S=1033$ thì phép dịch chuyển S thành 1833 là không hợp lệ, và S dịch chuyển thành 1733 là hợp lệ);
- c. Số các bước dịch chuyển là ít nhất.

Ví dụ số các phép dịch chuyển ít nhất để $S = 1033$ thành $T = 8179$ là 6 bao gồm các phép dịch chuyển như sau:

$8179 \rightarrow 8779 \rightarrow 3779 \rightarrow 3739 \rightarrow 3733 \rightarrow 1733 \rightarrow 1033$.

Input:

- Dòng đầu tiên đưa vào số lượng test T ($T \leq 100$)
- Những dòng kế tiếp mỗi dòng đưa vào một test. Mỗi test là một bộ đôi S, T .

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
1033 8179	6
1033 8779	5

6.16. DI CHUYỂN TRONG MA TRẬN

Cho ma trận $A[M][N]$. Nhiệm vụ của bạn hãy tìm **số bước đi ít nhất** dịch chuyển từ vị trí $A[1][1]$ đến vị trí $A[M][N]$. Biết mỗi bước đi ta chỉ được phép dịch chuyển đến vị trí $A[i][j+A[i][j]]$ hoặc vị trí $A[i+A[i][j]][j]$ bên trong ma trận.

Input:

- Dòng đầu tiên đưa vào số lượng test T .
- Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất là hai số M, N ; phần thứ hai là các phần tử của ma trận $A[][]$; các số được viết cách nhau một vài khoảng trắng.
- $T, M, N, A[i][j]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq M, N, A[i][j] \leq 10^3$.

Output:

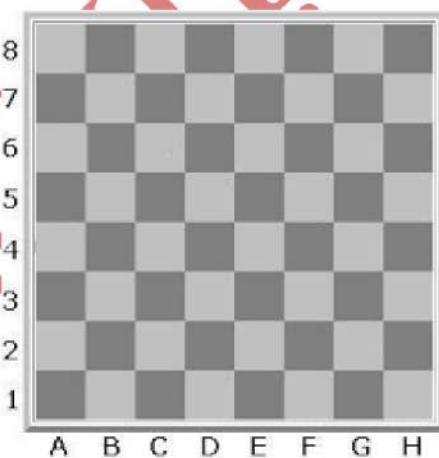
- Đưa ra kết quả mỗi test theo từng dòng. In ra -1 nếu không tìm được đáp án.

Ví dụ:

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

6.17. 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 y, 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	

6.18. 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.

Input:

- Dòng đầu ghi số bộ test (không quá 10). Mỗi bộ test gồm hai dòng:

- 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).

Output:

- Với mỗi test, in ra một số nguyên là đáp số của bài toán.

Ví dụ:

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

CHƯƠNG 7. ĐỒ THỊ

7.1. DFS TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều sâu bắt đầu tại đỉnh $u \in V$ ($DFS(u) = ?$)

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm $|E| + 1$ dòng: dòng đầu tiên đưa vào ba số $|V|, |E|$ tương ứng với số đỉnh và số cạnh của đồ thị, và u là đỉnh xuất phát; $|E|$ dòng tiếp theo đưa vào các bộ đôi $u \hat{v}$, $v \hat{v}$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 200$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán $DFS(u)$ của mỗi test theo khuôn dạng của ví dụ dưới đây.

Ví dụ:

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

7.2. BFS TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều rộng bắt đầu tại đỉnh $u \in V$ ($BFS(u) = ?$)

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số $|V|$, $|E|$, $u \in V$ tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi $u \in V$, $v \in V$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 200$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán $BFS(u)$ của mỗi test theo khuôn dạng của ví dụ dưới đây.

Ví dụ:

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

7.3. ĐƯỜNG ĐI THEO DFS VỚI ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh $s \in V$ đến đỉnh $t \in V$ trên đồ thị bằng thuật toán DFS.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.

- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào bốn số $|V|$, $|E|$, $s\hat{V}$, $t\hat{V}$ tương ứng với số đỉnh, số cạnh, đỉnh u, đỉnh v; Dòng tiếp theo đưa vào các bộ đôi $u\hat{V}$, $v\hat{V}$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra đường đi từ đỉnh s đến đỉnh t của mỗi test theo thuật toán DFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

Ví dụ:

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

7.4. ĐƯỜNG ĐI THEO BFS TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh $s\hat{V}$ đến đỉnh $t\hat{V}$ trên đồ thị bằng thuật toán BFS.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào bốn số $|V|$, $|E|$, $s\hat{V}$, $t\hat{V}$ tương ứng với số đỉnh, số cạnh, đỉnh u, đỉnh v; Dòng tiếp theo đưa vào các bộ đôi $u\hat{V}$, $v\hat{V}$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra đường đi từ đỉnh s đến đỉnh t của mỗi test theo thuật toán BFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

Ví dụ:

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

<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="padding: 2px;">1</td></tr> <tr><td style="padding: 2px;">6 9 1 6</td></tr> <tr><td style="padding: 2px;">1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6</td></tr> </table>	1	6 9 1 6	1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="padding: 2px;">1 2 5 6</td></tr> </table>	1 2 5 6
1					
6 9 1 6					
1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6					
1 2 5 6					

7.5. ĐẾM SỐ THÀNH PHẦN LIÊN THÔNG

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy tìm số thành phần liên thông của đồ thị.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi uV, vV tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra số thành phần liên thông của đồ thị.

Ví dụ:

Input:	Output:			
<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="padding: 2px;">1</td></tr> <tr><td style="padding: 2px;">5 6</td></tr> <tr><td style="padding: 2px;">1 2 1 3 2 3 3 4 3 5 4 5</td></tr> </table>	1	5 6	1 2 1 3 2 3 3 4 3 5 4 5	1
1				
5 6				
1 2 1 3 2 3 3 4 3 5 4 5				

7.6. TÌM SỐ THÀNH PHẦN LIÊN THÔNG VỚI BFS

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy tìm số thành phần liên thông của đồ thị bằng thuật toán BFS.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|$, $|E|$ tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi uV, vV tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra số thành phần liên thông của đồ thị bằng thuật toán BFS.

Ví dụ:

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

7.7. KIỂM TRA TÍNH LIÊN THÔNG MẠNH

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có liên thông mạnh hay không?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|$, $|E|$ tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi u, v tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra “YES”, hoặc “NO” theo từng dòng tương ứng với test là liên thông mạnh hoặc không liên thông mạnh.

Ví dụ:

Input:	Output:

1	
6 9	YES
1 2 2 4 3 1 3 2 3 5 4 3 5 4 5 6 6 3	

7.8. SỐ LƯỢNG HÒN ĐẢO

Cho một bản đồ kích thước $N \times M$ được mô tả bằng ma trận $A[][]$. $A[i][j] = 1$ có nghĩa vị trí (i, j) là nồi trên biển. 2 vị trí (i, j) và (x, y) được coi là liền nhau nếu 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 kỳ 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 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	5

7.9. LIỆT KÊ ĐỈNH TRỤ

Cho đồ thị vô hướng liên thông $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy đưa ra tất cả các đỉnh trụ của đồ thị?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi u, v tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra danh sách các đỉnh trụ của mỗi test theo từng dòng.

Ví dụ:

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

7.10. LIỆT KÊ CẠNH CẦU

Cho đồ thị vô hướng liên thông $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy đưa ra tất cả các cạnh cầu của đồ thị?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm $|E| + 1$ dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh và số cạnh; $|E|$ dòng tiếp theo đưa vào các bộ đôi u, v tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra danh sách các cạnh cầu của mỗi test theo từng dòng. In ra đáp án theo thứ tự từ điển, theo dạng “a b ...” với $a < b$.

Ví dụ:

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

7.11. KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ VÔ HƯỚNG

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi u, v tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2;$

Output:

- Đưa ra YES hoặc “NO” kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

Ví dụ:

Input:	Output:
1	
6 9	YES

1 2 1 3 2 3 2 5 3 4 3 5 4 5 4 6 5 6	
-------------------------------------	--

7.12. KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi u, v tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2;$

Output:

- Đưa ra YES hoặc “NO” kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

Ví dụ:

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

7.13. KIỂM TRA CHU TRÌNH TRÊN ĐỒ THỊ CÓ HƯỚNG VỚI DFS

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Sử dụng thuật toán DFS, hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.

- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|$, $|E|$ tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi uV , vV tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra YES hoặc “NO” kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

Ví dụ:

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

7.14. 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	
4	
1 2	
1 3	YES
2 4	NO
4	
1 2	
1 3	
2 3	

7.15. LIỆT KÊ ĐỈNH TRỤ

Cho đồ thị vô hướng liên thông $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy đưa ra tất cả các đỉnh trụ của đồ thị?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh và số cạnh; Dòng tiếp theo đưa vào các bộ đôi uV, vV tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra danh sách các đỉnh trụ của mỗi test theo từng dòng.

Ví dụ:

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

7.16. KIỂM TRA CHU TRÌNH

Cho đồ thị vô hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy kiểm tra xem đồ thị có tồn tại chu trình hay không?

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh, số cạnh của đồ thị; Dòng tiếp theo đưa vào các bộ đôi uV, vV tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra YES hoặc “NO” kết quả test theo từng dòng tương ứng với đồ thị tồn tại hoặc không tồn tại chu trình.

Ví dụ:

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

7.17. CHUYỂN DANH SÁCH KÈ SANG MA TRẬN KÈ

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

Hãy biểu diễn đồ thị bằng ma trận kề.

Input: 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 .

Output: 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	

7.18. 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.

Input: 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.

Output: 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.

Ví dụ:

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

7.19. DFS TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều sâu bắt đầu tại đỉnh $u \in V$ ($DFS(u) = ?$)

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số $|V|$, $|E|$, $u \in V$ tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi $u \in V$, $v \in V$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 200$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán $DFS(u)$ của mỗi test theo khuôn dạng của ví dụ dưới đây.

Ví dụ:

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

7.20. BFS TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy viết thuật toán duyệt theo chiều rộng bắt đầu tại đỉnh $u \in V$ ($BFS(u) = ?$)

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào ba số $|V|$, $|E|$, $u \in V$ tương ứng với số đỉnh, số cạnh và đỉnh bắt đầu duyệt; Dòng tiếp theo đưa vào các bộ đôi $u \in V$, $v \in V$ tương ứng với một cạnh của đồ thị.

- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 200; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2;$

Output:

- Đưa ra danh sách các đỉnh được duyệt theo thuật toán BFS(u) của mỗi test theo khuôn dạng của ví dụ dưới đây.

Ví dụ:

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

7.21. ĐƯỜNG ĐI THEO DFS VỚI ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh $s \in V$ đến đỉnh $t \in V$ trên đồ thị bằng thuật toán DFS.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào bốn số $|V|, |E|, s \in V, t \in V$ tương ứng với số đỉnh, số cạnh, đỉnh u , đỉnh v ; Dòng tiếp theo đưa vào các bộ đôi $u \in V, v \in V$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq |V| \leq 10^3; 1 \leq |E| \leq |V|(|V|-1)/2;$

Output:

- Đưa ra đường đi từ đỉnh s đến đỉnh t của mỗi test theo thuật toán DFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

Ví dụ:

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

1 2 2 5 3 1 3 2 3 5 4 3 5 4 5 6 6 4	
-------------------------------------	--

7.22. ĐƯỜNG ĐI THEO BFS TRÊN ĐỒ THỊ CÓ HƯỚNG

Cho đồ thị có hướng $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh. Hãy tìm đường đi từ đỉnh $u \in V$ đến đỉnh $v \in V$ trên đồ thị bằng thuật toán BFS.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào bốn số $|V|$, $|E|$, $s \in V$, $t \in V$ tương ứng với số đỉnh, số cạnh, đỉnh u , đỉnh v ; $|E|$. Dòng tiếp theo đưa vào các bộ đôi $u \in V$, $v \in V$ tương ứng với một cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra đường đi từ đỉnh s đến đỉnh t của mỗi test theo thuật toán BFS của mỗi test theo khuôn dạng của ví dụ dưới đây. Nếu không có đáp án, in ra -1.

Ví dụ:

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

7.23. KIỂM TRA ĐƯỜNG ĐI

Cho đồ thị vô hướng 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” nếu 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	

7.24. ĐƯỜNG ĐI HAMILTON

Đường đi đơn trên đồ thị có hướng hoặc vô hướng đi qua tất cả các đỉnh của đồ thị mỗi đỉnh đúng một lần được gọi là đường đi Hamilton. Cho đồ thị vô hướng $G = \langle V, E \rangle$, hãy kiểm tra xem đồ thị có đường đi Hamilton hay không?

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.

- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào hai số V, E tương ứng với số đỉnh, số cạnh của đồ thị; phần thứ hai đưa vào các cạnh của đồ thị.
- T, V, E thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq V \leq 10$; $1 \leq E \leq 15$.

Output:

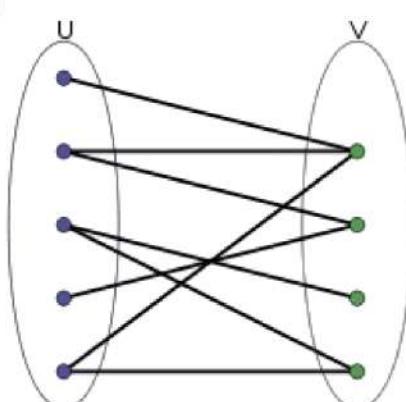
- Đưa ra 1 hoặc 0 tương ứng với test có hoặc không có đường đi Hamilton theo từng dòng.

Ví dụ:

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

7.25. ĐỒ THỊ HAI PHÍA

Đồ thị hai phia 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 phia 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	

7.26. KẾT BẠN

Trường học X có N sinh viên, trong đó có M cặp là bạn bè của nhau. Bạn của bạn cũng là bạn, tức là nếu A là bạn của B, B là bạn của C thì A và C cũng là bạn bè của nhau.

Các bạn hãy xác định xem số lượng sinh viên nhiều nhất trong một nhóm bạn là bao nhiêu?

Input:

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

Mỗi test bắt đầu bởi 2 số nguyên N và M ($N, M \leq 100\,000$).

M dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v ($u \# v$) cho biết sinh viên u là bạn của sinh viên v.

Output:

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

Ví dụ:

Input:	Output
2	
3 2	
1 2	
2 3	
10 12	
1 2	
3 1	
3 4	
5 4	3
3 5	7
4 6	
5 2	
2 1	
7 1	
1 2	
9 10	
8 9	

7.27. MẠNG XÃ HỘI

Tý đang xây dựng một mạng xã hội và mời các bạn của mình dùng thử. Bạn của bạn cũng là bạn. Vì vậy, Tý muốn mạng xã hội của mình là hoàn hảo, tức với mọi bộ ba X, Y, Z, nếu X kết bạn với Y, Y kết bạn với Z thì X và Z cũng phải là bạn bè của nhau trên mạng xã hội.

Các bạn hãy xác định xem mạng xã hội hiện tại của Tý có là hoàn hảo hay không? Nếu có hãy in ra “YES”, “NO” trong trường hợp ngược lại.

Input:

- Dòng đầu tiên là số lượng bộ test T ($T \leq 20$).
- Mỗi test bắt đầu bởi 2 số nguyên N và M ($N, M \leq 100\,000$).
- M dòng tiếp theo, mỗi dòng gồm 2 số nguyên u, v ($u \neq v$) cho biết u và v là kết bạn với nhau trên mạng xã hội của Tý.

Output:

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

Ví dụ:

Input:	Output
3	YES
4 3	NO
1 3	YES
3 4	
1 4	
4 4	
3 1	
2 3	
3 4	
1 2	
10 4	
4 3	
5 10	

8	9
1	2

7.28. 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 ($1 \leq N \leq 10^3$, $1 \leq M \leq 10^5$, $1 \leq u \leq N$).
- 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, 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	
4 4 2	2 1
1 2	1 3
1 3	3 4
2 4	-1
3 4	

4 2 2	
1 2	
3 4	

7.29. 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 ($1 \leq N \leq 10^3$, $1 \leq M \leq 10^5$, $1 \leq u \leq N$).
- 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, 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	2 1
1 2	2 4
1 3	1 3
2 4	-1
3 4	

4 2 2	
1 2	
3 4	

7.30. DIJKSTRA

Cho đồ thị có trọng số không âm $G = \langle V, E \rangle$ được biểu diễn dưới dạng danh sách cạnh trọng số. Hãy viết chương trình tìm đường đi ngắn nhất từ đỉnh $u \in V$ đến tất cả các đỉnh còn lại trên đồ thị.

Input:

- Dòng đầu tiên đưa vào T là số lượng bộ test.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm $|E|+1$ dòng: dòng đầu tiên đưa vào hai số $|V|, |E|$ tương ứng với số đỉnh và $u \in V$ là đỉnh bắt đầu; $|E|$ dòng tiếp theo mỗi dòng đưa vào bộ ba $u \in V, v \in V, w$ tương ứng với một cạnh cùng với trọng số cạnh của đồ thị.
- $T, |V|, |E|$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq |V| \leq 10^3$; $1 \leq |E| \leq |V|(|V|-1)/2$;

Output:

- Đưa ra kết quả của mỗi test theo từng dòng. Kết quả mỗi test là trọng số đường đi ngắn nhất từ đỉnh u đến các đỉnh còn lại của đồ thị theo thứ tự tăng dần các đỉnh.

Ví dụ:

Input:	Output:
1 9 12 1 1 2 4 1 8 8 2 3 8 2 8 11 3 4 7	0 4 12 19 26 16 18 8 14

3	6	4
3	9	2
4	5	9
4	6	14
5	6	10
6	7	2
6	9	6

7.31. FLOYD

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	

7.32. DI CHUYỂN TRÊN 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

7.33. KRUSKAL

Cho đồ thị vô hướng có trọng số $G = \langle V, E, W \rangle$. Nhiệm vụ của bạn là hãy xây dựng một cây khung nhỏ nhất của đồ thị bằng thuật toán Kruskal.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai phần: phần thứ nhất đưa vào hai số V, E tương ứng với số đỉnh và số cạnh của đồ thị; phần thứ 2 đưa vào E cạnh của đồ thị, mỗi cạnh là một bộ 3: đỉnh đầu, đỉnh cuối và trọng số của cạnh.
- T, S, D thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq V \leq 100$; $1 \leq E, W \leq 10000$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output

2	4
3 3	5
1 2 5	
2 3 3	
1 3 1	
2 1	
1 2 5	

7.34. 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:

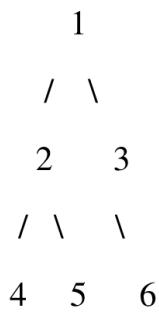
Input:	Output
1	3.414214
3	
1.0 1.0	
2.0 2.0	

THƯ VIỆN PTIT

CHƯƠNG 8. CÂY NHỊ PHÂN

8.1. DUYỆT CÂY 1

Cho phép duyệt cây nhị phân Inorder và Preorder, hãy đưa ra kết quả phép duyệt Postorder của cây nhị phân. Ví dụ với cây nhị phân có các phép duyệt cây nhị phân của cây dưới đây:



Inorder : 4 2 5 1 3 6

Preorder: 1 2 4 5 3 6

Postorder : 4 5 2 6 3 1

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng node; dòng tiếp theo đưa vào N số theo phép duyệt Inorder; dòng cuối cùng đưa vào N số là kết quả của phép duyệt Preorder; các số được viết cách nhau một vài khoảng trắng.
- T, N, node thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 1000$; $1 \leq$ giá trị node $\leq 10^4$;

Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

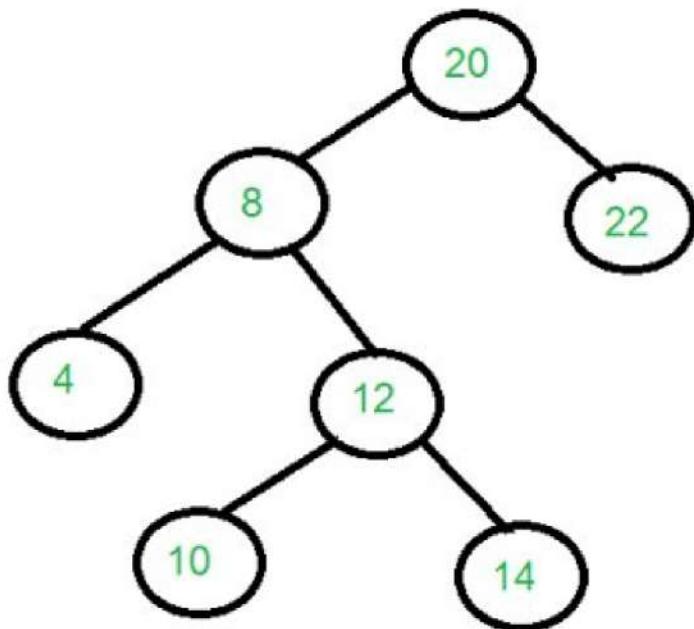
Ví dụ:

Input	Output
1	4 5 2 6 3 1

6	
4 2 5 1 3 6	
1 2 4 5 3 6	

8.2. DUYỆT CÂY THEO MỨC

Cho cây nhị phân, nhiệm vụ của bạn là duyệt cây theo Level-order. Phép duyệt level-order trên cây là phép thăm node theo từng mức của cây. Ví dụ với cây dưới đây sẽ cho ta kết quả của phép duyệt level-order: 20 8 22 4 12 10 14.



Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

- Đưa ra kết quả phép duyệt level-order theo từng dòng.

Ví dụ:

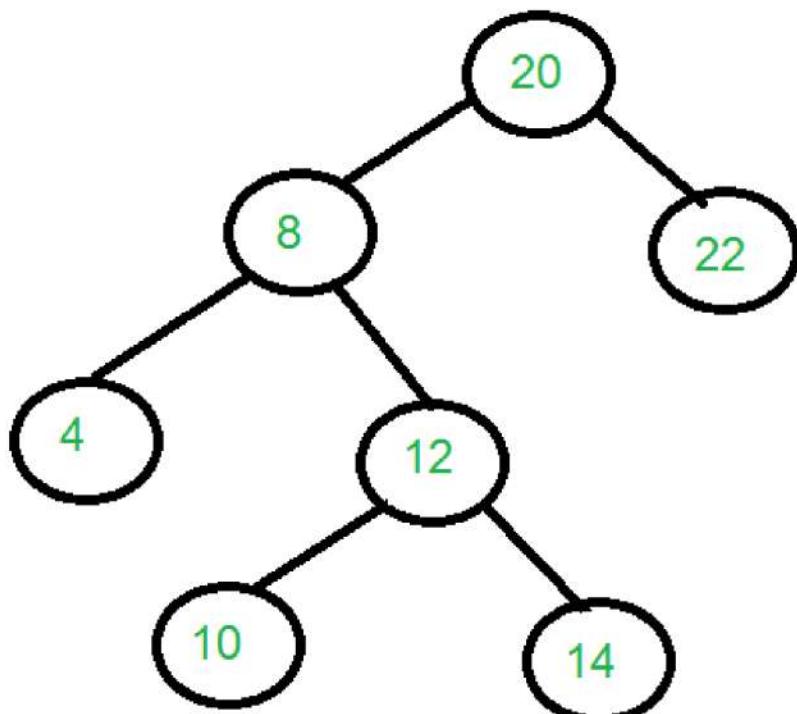
Input	Output
2	
2	1 3 2
1 2 R 1 3 L	
4	10 20 30 40 60
10 20 L 10 30 R 20 40 L 20 60 R	

8.3. DUYỆT CÂY 2

Cho hai mảng là phép duyệt Inorder và Level-order, nhiệm vụ của bạn là xây dựng cây nhị phân và đưa ra kết quả phép duyệt Postorder. Level-order là phép duyệt theo từng mức của cây. Ví dụ như cây dưới đây ta có phép Inorder và Level-order như dưới đây:

Inorder : 4 8 10 12 14 20 22

Level order: 20 8 22 4 12 10 14



Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng node; dòng tiếp theo đưa vào N số là phép duyệt Inorder; dòng cuối cùng đưa vào N số là phép duyệt Level-order; các số được viết cách nhau một vài khoảng trắng.
- T, N, node thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq A[i] \leq 10^4$;

Output:

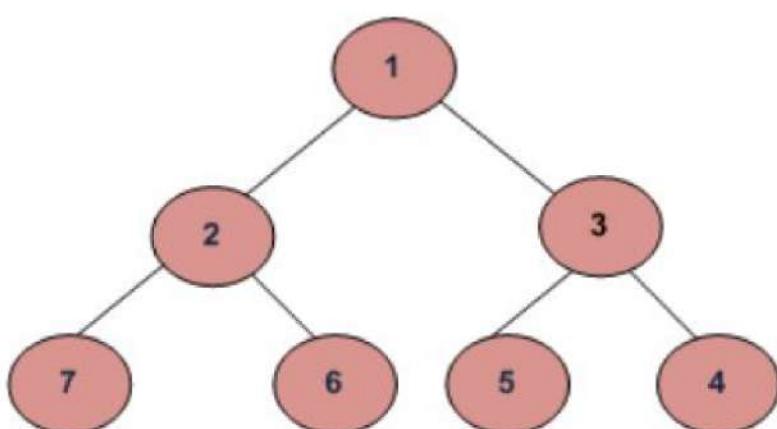
- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

Ví dụ:

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

8.4. DUYỆT CÂY KIỂU XOẮN ÔC

Cho cây nhị phân, nhiệm vụ của bạn là duyệt cây theo xoắn ốc (spiral-order). Phép. Ví dụ với cây dưới đây sẽ cho ta kết quả của phép duyệt spiral-order: 1 2 3 4 5 6 7.



Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

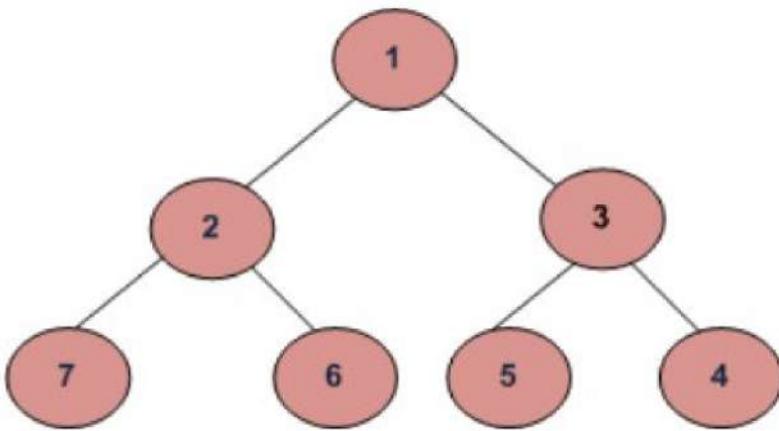
- Đưa ra kết quả phép duyệt level-order theo từng dòng.

Ví dụ:

Input	Output
2	
2	
1 2 R 1 3 L	1 3 2
4	
10 20 L 10 30 R 20 40 L 20 60 R	10 0 30 60 40

8.5. KIỂM TRA NODE LÁ

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem tất cả các node lá của cây có cùng một mức hay không? Ví dụ với cây dưới đây sẽ cho ta kết quả là Yes.



Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2 2 1 2 R 1 3 L 4 10 20 L 10 30 R 20 40 L 20 60 R	1 0

8.6. CÂY NHỊ PHÂN HOÀN HẢO

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có phải là một cây hoàn hảo hay không (perfect tree)? Một cây nhị phân được gọi là cây hoàn hảo nếu tất

cả các node trung gian của nó đều có hai node con và tất cả các node lá đều có cùng một mức.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3	
6	
10 20 L 10 30 R 20 40 L 20 50 R 30 60 L 30 70 R	Yes
2	Yes
18 15 L 18 30 R	No
5	
1 2 L 2 4 R 1 3 R 3 5 L 3 6 R	

8.7. CÂY NHỊ PHÂN ĐỦ

Cho cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có phải là một cây đủ hay không (full binary tree)? Một cây nhị phân được gọi là cây đủ nếu tất cả các node trung gian của nó đều có hai node con.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x),

trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.

- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	1
4	0
1 2 L 1 3 R 2 4 L 2 5 R	
3	
1 2 L 1 3 R 2 4 L	

8.8. CÂY NHỊ PHÂN BẰNG NHAU

Cho hai cây nhị phân, nhiệm vụ của bạn là kiểm tra xem cây nhị phân có giống nhau hay không?

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái của mỗi cây; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

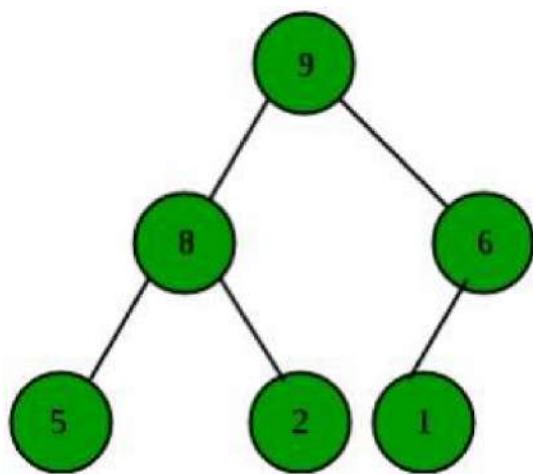
Input	Output

2	
2	
1 2 L 1 3 R	
2	
1 2 L 1 3 R	
2	
1 2 L 1 3 R	
2	
1 3 L 1 2 R	

1	
0	

8.9. TỔNG NODE LÁ BÊN PHẢI

Cho cây nhị phân, nhiệm vụ của bạn là tính tổng của tất cả các node lá bên phải trên cây? Ví dụ với cây dưới đây ta có kết quả là 2.



Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

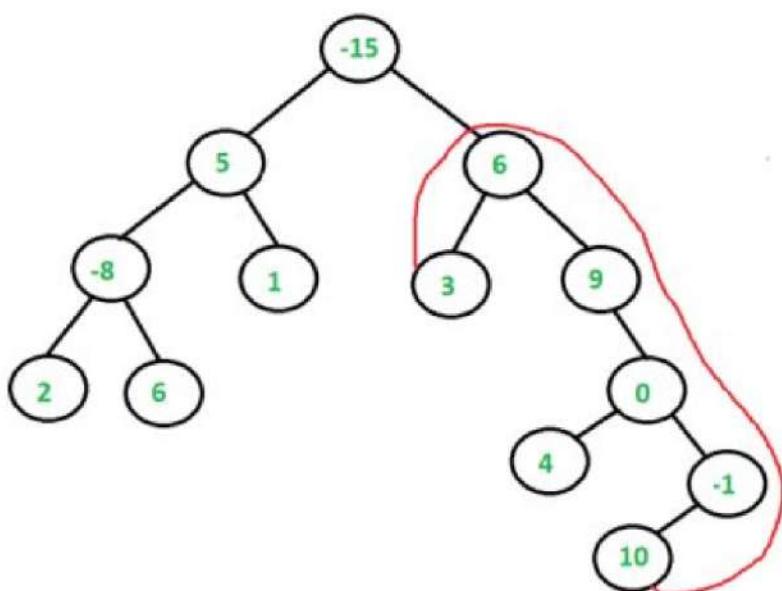
- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
2	
1 2 L 1 3 R	
5	
10 20 L 10 30 R 20 40 L 20 60 R 30 90 L	3 60

8.10. TỔNG LỚN NHẤT

Cho cây nhị phân có giá trị mỗi node là một số, nhiệm vụ của bạn là tìm tổng lớn nhất từ một node lá này sang một node lá khác? Ví dụ với cây dưới đây ta có tổng lớn nhất là 27.



Input:

- Dòng đầu tiên đưa vào số lượng test T.

- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.
- T, N, u, v, thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq u, v \leq 10^4$;

Output:

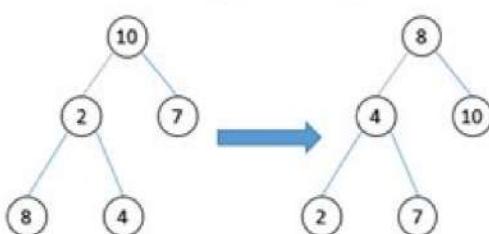
- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
1 12 -15 5 L -15 6 R 5 -8 L 5 1 R -8 2 L -8 -3 R 6 3 L 6 9 R 9 0 R 0 4 L 0 -1 R -1 10 L	27

8.11. BIẾN ĐỔI SANG CÂY NHỊ PHÂN TÌM KIẾM

Cho cây nhị phân, nhiệm vụ của bạn là dịch chuyển cây nhị phân thành cây nhị phân tìm kiếm. Phép dịch chuyển phải bảo toàn được cấu trúc cây nhị phân ban đầu. Ví dụ dưới đây sẽ minh họa phép dịch chuyển:



Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 3 dòng: dòng đầu tiên đưa vào số N là số lượng cạnh của cây; dòng tiếp theo đưa vào N bộ ba (u, v, x), trong đó u là node cha, v là node con, x= R nếu v là con phải, x=L nếu v là con trái; u, v, x được viết cách nhau một vài khoảng trắng.

- T, N, u, v , thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq N \leq 10^3; 1 \leq u, v \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng là phép duyệt Inorder của cây tìm kiếm.

Ví dụ:

Input	Output
2 2 1 2 R 1 3 L 4 10 20 L 10 30 R 20 40 L 20 60 R	1 2 3 10 20 30 40 60

8.12. DUYỆT CÂY NHỊ PHÂN TÌM KIẾM 1

Cho mảng $A[]$ gồm N node là biểu diễn phép duyệt theo thứ tự giữa (Preorder) của cây nhị phân tìm kiếm. Nhiệm vụ của bạn là đưa ra phép duyệt theo thứ tự sau của cây nhị phân tìm kiếm.

Input:

- Dòng đầu tiên đưa vào số lượng test T .
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng node; dòng tiếp theo đưa vào N số $A[i]$; các số được viết cách nhau một vài khoảng trắng.
- $T, N, node$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq N \leq 10^3; 1 \leq A[i] \leq 10^4$;

Output:

- Đưa ra kết quả phép duyệt Postorder theo từng dòng.

Ví dụ:

Input	Output

2	
5	35 30 100 80 40
40 30 35 80 100	35 32 30 120 100 90 80 40
8	
40 30 32 35 80 90 100 120	

8.13. XÂY DỰNG LẠI CÂY NHỊ PHÂN TÌM KIẾM

Cho một mảng là phép duyệt level-order của cây nhị phân tìm kiếm. Nhiệm vụ của bạn là xây dựng lại cây nhị phân tìm kiếm bảo toàn được cấu trúc cây nhị phân ban đầu.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm dòng: dòng đầu tiên đưa vào số N là số lượng node của cây tìm kiếm; dòng tiếp theo đưa vào phép duyệt level-order của cây tìm kiếm; các số được viết cách nhau một vài khoảng trắng.
- T, N, node thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq \text{node} \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng là phép duyệt trước (preOrder) của cây tìm kiếm.

Ví dụ:

Input	Output
2 9 7 4 12 3 6 8 1 5 10 6 1 3 4 6 7 8	7 4 3 1 6 5 12 8 10 1 3 4 6 7 8

8.14. DUYỆT CÂY NHỊ PHÂN TÌM KIẾM 2

Cho một mảng A[] gồm N phần tử biểu diễn phép duyệt preorder của cây nhị phân tìm kiếm. Nhiệm vụ của bạn là đưa ra phép duyệt postorder của cây nhị phân tìm kiếm.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng node của cây tìm kiếm; dòng tiếp theo đưa vào phép duyệt preorder của cây tìm kiếm; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq A[i] \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng là phép duyệt postorder của cây tìm kiếm.

Ví dụ:

Input	Output
2	
5	
40 30 35 80 100	35 30 100 80 40
8	
40 30 32 35 80 90 100 120	35 32 30 120 100 90 80 40

8.15. KIỂM TRA CÂY NHỊ PHÂN TÌM KIẾM

Cho một mảng A[] gồm N phần tử. Nhiệm vụ của bạn là đưa ra 1 nếu mảng A[] biểu diễn phép duyệt inorder của cây nhị phân tìm kiếm, ngược lại đưa ra 0.

Input:

- Dòng đầu tiên đưa vào số lượng test T.
- Những dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm 2 dòng: dòng đầu tiên đưa vào số N là số lượng node của cây tìm kiếm; dòng tiếp theo đưa vào N số A[i]; các số được viết cách nhau một vài khoảng trắng.
- T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^3$; $1 \leq A[i] \leq 10^4$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
3 5 10 20 30 40 50 6 90 80 100 70 40 30 3 1 1 2	1 0 0

8.16. NODE LÁ CỦA CÂY NHỊ PHÂN TÌM KIẾM

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preorder) 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 ($N \leq 1000$). 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.

Ví dụ:

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	

8.17. NODE TRUNG GIAN CỦA CÂY NHỊ PHÂN TÌM KIẾM

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preorder) 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 ($N \leq 1000$). 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.

Ví dụ:

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

8.18. ĐỘ SÂU CÂY NHỊ PHÂN TÌM KIẾM

Cho dãy số gồm N số là phép duyệt theo thứ tự trước (Preorder) 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 ($N \leq 1000$). 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.

Ví dụ:

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

8.19. CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG 1

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số $A[] = (a_0, a_1, \dots, a_{n-1})$. Đưa ra nội dung node gốc của cây tìm kiếm cân bằng. Ví dụ với dãy $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$ ta sẽ có cây nhị phân tìm kiếm cân bằng với node gốc là 33.

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 ($N \leq 10^6$). Dòng tiếp theo là N số của mảng $A[]$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
11	30
40 28 45 38 33 15 25 20 23 35 30	5
10	
1 2 3 4 5 6 7 8 9 10	

8.20. CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG 2

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số $A[] = \{a_0, a_1, \dots, a_{n-1}\}$. Đưa ra phép duyệt theo thứ tự trước (preorder) của cây tìm kiếm cân bằng. Ví dụ với dãy $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$ ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với node gốc là 33 : 33, 25, 20, 15, 23, 28, 30, 40, 38, 35, 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 ($N \leq 10^6$). Dòng tiếp theo là N số của mảng $A[]$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input	Output
2	
11	30 23 15 20 25 28 38 33 35 40 45
40 28 45 38 33 15 25 20 23 35 30	5 2 1 3 4 8 6 7 9 10
10	

8.21. DUYỆT SAU CÂY TÌM KIẾM CÂN BẰNG

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số $A[] = \{a_0, a_1, \dots, a_{n-1}\}$. Dựa ra phép duyệt theo thứ tự sau (post-order) của cây tìm kiếm cân bằng. Ví dụ với dãy $A[] = \{40, 28, 45, 38, 33, 15, 25, 20, 23, 35, 30\}$ ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với node gốc là 33 : 15, 23, 20, 30, 28, 25, 35, 38, 45, 40, 33.

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 ($N \leq 10^6$). Dòng tiếp theo là N số của mảng $A[]$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	20 15 28 25 23 35 33 45 40 38 30
11	1 4 3 2 7 6 10 9 8 5
40 28 45 38 33 15 25 20 23 35 30	
10	
1 2 3 4 5 6 7 8 9 10	

8.22. ĐẾM SỐ NODE LÁ CÂY NHỊ PHÂN TÌM KIẾM CÂN BẰNG

Hãy xây dựng một cây nhị phân tìm kiếm cân bằng từ dãy số $A[] = \{a_0, a_1, \dots, a_{n-1}\}$. Đếm số node lá của cây nhị phân tìm kiếm cân bằng. Ví dụ với dãy $A[] = \{40, 28, 45,$

$\{38, 33, 15, 25, 20, 23, 35, 30\}$ ta sẽ có phép duyệt theo thứ tự trước của cây nhị phân tìm kiếm cân bằng với số node lá là $5 : 15, 23, 30, 35, 45$.

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 ($N \leq 1000$). Dòng tiếp theo là N số của mảng $A[]$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output
2	5
11	4
40 28 45 38 33 15 25 20 23 35 30	
10	
1 2 3 4 5 6 7 8 9 10	

TÀI LIỆU THAM KHẢO

- [1]. Donald E. Knuth., “The Art of Computer Programming”, Addison-Wesley, Vol 1, 2, 3,4,5,
- [2]. <https://www.geeksforgeeks.org/>
- [3]. <https://codeforces.com/>

THƯ VIỆN PTIT