

Câu 1. Tên bài: TOURISM.CPP

Cho n thành phố đánh số từ 1 đến n và m tuyến đường giao thông hai chiều giữa chúng, mạng lưới giao thông này được cho bởi bảng C cấp $n \times n$, ở đây $C_{ij} = C_{ji}$ = Chi phí đi đoạn đường trực tiếp từ thành phố i đến thành phố j . Giả thiết rằng $C_{ii} = 0$ với mọi i , $C_{ij} = +\infty$ nếu không có 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 cho người đó hành trình với chi phí ít nhất. Bài toán đó gọi là bài toán người du lịch hay bài toán hành trình của một thương gia (Traveling Salesman).

Dữ liệu: vào từ file văn bản **TOURISM.INP**

+ Dòng 1: Chứa số thành phố n ($1 \leq n \leq 15$) và số tuyến đường m trong mạng lưới giao thông

+ m dòng tiếp theo, mỗi dòng ghi số hiệu hai thành phố có đường đi trực tiếp và chi phí đi trên quãng đường đó (chi phí này là số nguyên dương ≤ 100)

Kết quả: ghi ra file văn bản **TOURISM.OUT**, ghi chi phí ít nhất tìm được.

Nếu không tồn tại đường đi, in ra -1.

Ví dụ:

TOURISM.INP	TOURISM.OUT
4 6	6
1 2 3	
1 3 2	
1 4 1	
2 3 1	
2 4 2	
3 4 4	

Câu 2. Tên bài: CAITUL.CPP

Trong siêu thị có n gói hàng, gói hàng thứ i có trọng lượng là $W_i \leq 100$ và giá trị $V_i \leq 100$. Một Tên trộm đột nhập vào siêu thị, sức của tên trộm

không thể mang được trọng lượng vượt quá M ($M \leq 100$). Hỏi tên trộm sẽ lấy đi những gói hàng nào để được tổng giá trị lớn nhất.

Dữ liệu: vào từ file **CAITUI.INP** gồm

- + Dòng đầu tiên gồm 2 số nguyên dương n và m ($n \leq 40, m \leq 100$)
- + n dòng tiếp theo, mỗi dòng chứa 2 số nguyên W_i và V_i ($W_i, V_i \leq 100$) lần lượt là trọng lượng và giá trị của gói hàng thứ i .

Kết quả: ghi ra file **CAITUI.OUT** là giá trị lớn nhất tên trộm lấy được.

CAITUI.INP	CAITUI.OUT
3 4	10
1 4	
2 5	
3 6	

Câu 3. Tên bài: NGOAC.CPP

Có thể định nghĩa khái niệm dãy ngoặc đúng dưới dạng đệ quy như sau:

1. () là dãy ngoặc đúng
2. C là dãy ngoặc đúng nếu $C = (A)$ hay $C = AB$ với A, B là các dãy ngoặc đúng.

Ví dụ dãy ngoặc đúng: (), (()), (), (())()

Ví dụ dãy ngoặc sai:), (((, ()((,)))),)()()

Yêu cầu: Bạn hãy viết chương trình liệt kê tất cả các dãy ngoặc đúng có chiều dài n (n chẵn)

Dữ liệu: vào từ file **NGOAC.INP** gồm 1 số nguyên dương n (n chẵn, $n \leq 20$)

Kết quả: ghi ra file **NGOAC.OUT**

+ Trong m dòng đầu tiên, mỗi dòng liệt kê một dãy ngoặc đúng chiều dài n . Các dãy được liệt kê theo thứ tự từ điển: '(' < ')

+ Dòng cuối cùng là số m . (m là số lượng các dãy ngoặc đúng có chiều dài n)

NGOAC.INP	NGOAC.OUT
4	(()) () 2

Câu 4. Tên bài: TRES.CPP

Hôm nay, vì sinh nhật Tí, nên Tí mời mọi người đi ăn ở nhà hàng. Mọi việc chuẩn bị đã gần xong nhưng còn một vấn đề làm Tí khá nhức đầu, đó là làm sao chọn được những món ăn mà mọi người cùng thích. Nhà hàng có M món ăn khác nhau và thú vị ở chỗ là mỗi món ăn rất nhiều nên có thể đủ cho bao nhiêu người cũng được, vì thế vấn đề là gọi món nào chứ không phải mỗi món gọi bao nhiêu.

Có tất cả N người đến dự tiệc sinh nhật (bao gồm cả Tí trong đó). Tí đã tìm hiểu được danh sách những món ăn yêu thích của từng người và muốn rằng đối với mỗi người phải có ít nhất 2 món mà họ thích. Tuy nhiên sau khi ăn xong còn nhiều tiết mục hấp dẫn khác nên bạn cũng muốn rằng bất kỳ ai cũng không có quá 2 món ăn yêu thích trong danh sách được đặt trước. Và vấn đề cuối cùng, đây là tiền của bố mẹ nên cũng không nên tiêu xài quá đáng.

Yêu cầu: Hãy cho biết số tiền ít nhất phải trả để gọi một thực đơn thỏa mãn các yêu cầu trên.

Dữ liệu: vào từ file **TRES.INP**

- + Dòng đầu tiên chứa hai số M, N ($M \leq 30; N \leq 10$)
- + Dòng thứ hai chứa M số P_i là giá của món thứ i .
- + Trong N dòng cuối cùng, dòng thứ k ghi danh sách các món yêu thích của người thứ k .

Kết quả: ghi ra file **TRES.OUT** gồm một số duy nhất là kết quả của bài toán, hoặc in ra -1 nếu không có cách gọi món nào thỏa mãn.

TRES.INP	TRES.OUT
5 3	450
100 150 300 425 200	
1 2 4	
1 3 4 5	
1 4 5	

Câu 5. Tên bài: MINE.CPP

Cho một bãi mìn kích thước $m*n$ ô vuông, trên một ô có thể có chứa một quả mìn hoặc không, để biểu diễn bản đồ mìn đó, người ta có hai cách:

+ Cách 1: dùng bản đồ đánh dấu: sử dụng một lưới ô vuông kích thước $m*n$, trên đó tại ô (i, j) ghi số 1 nếu ô đó có mìn, ghi số 0 nếu ô đó không có mìn

+ Cách 2: dùng bản đồ mật độ: sử dụng một lưới ô vuông kích thước $m*n$, trên đó tại ô (i, j) ghi một số trong khoảng từ 0 đến 8 cho biết tổng số mìn trong các ô lân cận với ô (i, j) (ô lân cận với ô (i, j) là ô có chung với ô (i, j) ít nhất 1 đỉnh).

Giả thiết rằng hai bản đồ được ghi chính xác theo tình trạng mìn trên hiện trường. Về nguyên tắc, lúc cài bãi mìn phải vẽ cả bản đồ đánh dấu và bản đồ mật độ, tuy nhiên sau một thời gian dài, khi người ta muốn gỡ mìn ra khỏi bãi thì vấn đề hết sức khó khăn bởi bản đồ đánh dấu đã bị thất lạc !!. **Công việc của các lập trình viên là: Từ bản đồ mật độ, hãy tái tạo lại bản đồ đánh dấu của bãi mìn.**

Dữ liệu: vào từ file văn bản **MINE.INP** gồm

+ Dòng 1: Ghi 2 số nguyên dương m, n ($2 \leq m, n \leq 20$)

+ m dòng tiếp theo, dòng thứ i ghi n số trên hàng i của bản đồ mật độ theo đúng thứ tự từ trái qua phải.

Kết quả: ghi ra file văn bản **MINE.OUT**:

+ Dòng 1: ghi tổng số lượng mìn trong bãi

+ m dòng tiếp theo, dòng thứ i ghi n số trên hàng i của bản đồ đánh dấu theo đúng thứ tự từ trái qua phải.

MINE.INP	MINE.OUT
2 5	5
0 2 1 3 2	1 0 1 1 1
1 2 2 4 2	0 0 0 0 1