

Bài toán người đưa thư

Có N thành phố, các thành phố được đánh số hiệu từ 1 tới N . Biết $C(i, j)$ là số phút để đi từ thành phố i tới thành phố j . Các con đường nối giữa các thành phố là một chiều nên $C(i, j)$ không nhất thiết phải bằng $C(j, i)$.

Một người đưa thư cần phát thư cho N thành phố. Người đưa thư có thể xuất phát tại một thành phố bất kỳ, sau đó lần lượt di chuyển tới các thành phố chưa được phát thư để phát thư.

Yêu cầu: Bạn hãy xác định một thứ tự phát thư sao cho tổng thời gian di chuyển của người đưa thư là nhỏ nhất.

Input: đọc từ file **tsp.inp**

- Dòng đầu tiên chứa số nguyên dương N ($N \leq 15$) là số lượng thành phố.
- N dòng tiếp theo, dòng thứ i ($1 \leq i \leq N$) chứa N số nguyên không âm $C(i, 1), C(i, 2), \dots, C(i, N)$. Dữ liệu đảm bảo $C(i, i) = 0$.

Output: ghi ra file **tsp.out**

Dòng đầu tiên in ra số phút ít nhất để người đưa thư có thể phát thư cho N thành phố.

Dòng thứ hai chứa một hoán vị của N thành phố mô tả thứ tự phát thư tối ưu. Nếu có nhiều thứ tự phát thư tối ưu, in ra một cách bất kỳ.

Ví dụ:

tsp.inp	tsp.out	Giải thích
3 0 3 1 1 0 2 5 4 0	2 2 1 3	$C(2, 1) + C(1, 3) = 2$