

## BÀI TẬP CHUYÊN ĐỀ TIN HỌC

### CHƯƠNG TRÌNH HỌC

Chương trình học của một trường đại học có  $n$  môn đánh số từ 1 tới  $n$ , môn học thứ  $i$  cần học trong  $t_i$  học kỳ. Có  $m$  quan hệ giữa các môn học, quan hệ thứ  $j$  cho biết sinh viên bắt buộc phải học xong môn  $u_j$  trước khi bắt đầu học môn  $v_j$ . Chương trình đào tạo được cho hợp lý để sinh viên có thể hoàn thành hết tất cả các môn học, các học kỳ được đánh số từ 1 theo trình tự thời gian. Sinh viên có thể học nhiều môn trong cùng một học kỳ miễn là không vi phạm những mối quan hệ giữa các môn học.

**Yêu cầu:** Hãy lập một lịch học để sinh viên có thể hoàn thành hết tất cả các môn một cách nhanh nhất.

**Dữ liệu:** Vào từ tập tin văn bản **PROGRAM.INP**

- Dòng 1 chứa hai số nguyên dương  $n \leq 10^5; m \leq 2.10^5$
- Dòng 2 chứa  $n$  số nguyên dương  $t_1, t_2, \dots, t_n$  ( $t_i \leq 10^5, \forall i$ )
- $m$  dòng tiếp theo, dòng thứ  $j$  chứa hai số nguyên dương  $u_j, v_j$

**Kết quả:** Ghi ra tập tin văn bản **PROGRAM.OUT**

- Dòng 1: Ghi số học kỳ ít nhất để hoàn thành tất cả các môn.
- $n$  dòng tiếp theo, dòng thứ  $i$  ghi số hiệu học kỳ bắt đầu học môn  $i$ .

**Ví dụ:**

PROGRAM . INP	PROGRAM . OUT
5 6	12
5 4 3 2 1	8
5 2	4
5 1	1
3 2	8
3 4	1
2 1	
2 4	

## MẠNG GIAO THÔNG

Trong quốc gia mà chúng ta đang xét có  $n$  thành phố được nối với nhau bởi  $m$  đường 2 chiều, mỗi đường nối 2 thành phố (không nhất thiết khác nhau). Mạng lưới giao thông này đảm bảo từ một thành phố bất kỳ có thể tới được thành phố bất kỳ khác. Một số đường trong mạng có thu phí giao thông.

Để giảm bớt chi ngân sách Bộ GTVT lên phương án đóng cửa một số đường, chỉ giữ lại  $n - 1$  đường nhưng vẫn đảm bảo từ một thành phố bất kỳ có thể tới được thành phố bất kỳ khác và trong số đó có đúng  $k$  đường thu phí – đủ để bù ngân sách và không gây bất bình lớn trong xã hội.

**Yêu cầu:** Hãy đưa một phương án chỉ ra các đường cần giữ lại. Nếu không tồn tại phương án thích hợp thì đưa ra số  $-1$ .

**Dữ liệu:** Vào từ tập tin văn bản **ROADS.INP**

- Dòng đầu tiên chứa 3 số nguyên  $n, m, k$  ( $1 \leq k < n \leq 10^5; 1 \leq m \leq 2 \times 10^5$ ),
- Dòng thứ  $i$  trong  $m$  dòng sau chứa 3 số nguyên  $a_i, b_i, c_i$  cho biết đường thứ  $i$  nối 2 thành phố  $a_i$  và  $b_i$ ,  $c_i = 0$  – không thu phí,  $c_i = 1$  – có thu phí.

**Kết quả:** Ghi ra tập tin văn bản **ROADS.OUT** trên một dòng  $n - 1$  số nguyên xác định các con đường cần giữ lại hoặc một số  $-1$  nếu không có phương án.

**Ví dụ:**

ROADS . INP	ROADS . OUT
3 3 1 1 2 0 2 3 0 3 1 1	1 3

- Subtask 1 (40%):  $n, m \leq 1000$ .
- Subtask 2 (60%): không ràng buộc gì thêm.