



Môn: TIN HỌC

Thời gian: 180 phút (không kể thời gian giao đề)

Ngày thi thứ nhất: 05/01/2024

(Đề thi gồm 06 trang, 03 câu)

## TỔNG QUAN ĐỀ THI

	Tiêu đề	File chương trình	File dữ liệu	File kết quả
Câu 1	Ba đường truyền điện	THREE.*	THREE.INP	THREE.OUT
Câu 2	Cải thiện đánh giá	IMPEVAL.*	IMPEVAL.INP	IMPEVAL.OUT
Câu 3	Thu mua nông sản	FBUY.*	FBUY.INP	FBUY.OUT

Đấu \* được thay thế bởi PAS hoặc CPP tương ứng với ngôn ngữ lập trình Pascal hoặc C++.

Hãy lập trình giải các câu sau:

**Câu 1. Ba đường truyền điện** (7,0 điểm)

Quốc gia Alpha có một trang trại điện gió được quy hoạch bởi một bảng vuông  $N$  hàng và  $N$  cột. Các hàng được đánh số từ 1 tới  $N$  từ trên xuống dưới, các cột được đánh số từ 1 tới  $N$  từ trái sang phải. Trang trại có  $M$  trạm sản xuất điện gió được đánh số từ 1 tới  $M$ . Trạm thứ  $i$  ( $1 \leq i \leq M$ ) được đặt tại ô thuộc hàng  $R_i$ , cột  $C_i$  và có công suất sản xuất là  $W_i$  oát. Chủ trang trại mới ký hợp đồng cung cấp điện cho đối tác. Trang trại sẽ phải lắp ba đường truyền điện, mỗi đường truyền đi qua một hàng hoặc một cột trong bảng. Các đường truyền có thể cắt nhau nhưng không được trùng nhau. Tổng công suất cung cấp cho đối tác là tổng công suất của các trạm điện nằm trên ít nhất một trong ba đường truyền. Trang trại cần tìm ra cách lắp đặt ba đường truyền để tổng công suất cung cấp là lớn nhất có thể.

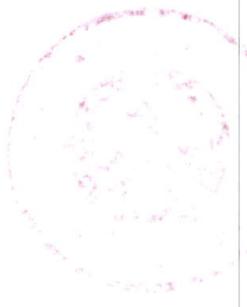
Ngoài ra, công ty có  $Q$  phương án điều chỉnh. Phương án điều chỉnh thứ  $j$  ( $1 \leq j \leq Q$ ) là tăng công suất của trạm  $T_j$  thêm một lượng  $D_j$  oát và giữ nguyên công suất  $W_i$  oát như hiện trạng ban đầu của mọi trạm  $i$  khác  $T_j$ . Lưu ý, các phương án là độc lập, nghĩa là các phương án đều chỉ áp dụng lên hiện trạng ban đầu của trang trại.

**Yêu cầu:** Hãy đưa ra tổng công suất lớn nhất có thể khi lắp ba đường truyền với hiện trạng ban đầu của trang trại và trong  $Q$  phương án điều chỉnh.

**Dữ liệu**

Vào từ file văn bản THREE.INP:

- Dòng đầu ghi một số nguyên dương  $\mathcal{T}$  là số lượng trường hợp test.
- Mỗi nhóm dòng trong số  $\mathcal{T}$  nhóm dòng tiếp theo mô tả một trường hợp test có cấu trúc như sau:
  - Dòng thứ nhất chứa ba số nguyên  $N$ ,  $M$  và  $Q$  lần lượt là kích thước bảng, số lượng trạm điện và số lượng phương án điều chỉnh ( $3 \leq N \leq 10^9$ ;  $3 \leq M \leq 10^5$ ;  $1 \leq Q \leq 10^5$ ).
  - Dòng thứ  $i$  trong số  $M$  dòng tiếp theo chứa ba số nguyên  $R_i$ ,  $C_i$  và  $W_i$  lần lượt là vị trí hàng, vị trí cột và công suất của trạm điện thứ  $i$ . Dữ liệu bảo đảm không có hai trạm nào đặt tại cùng một ô ( $1 \leq R_i, C_i \leq N$ ;  $1 \leq W_i \leq 10^9$ ).
  - Dòng thứ  $j$  trong số  $Q$  dòng tiếp theo chứa hai số nguyên  $T_j$  và  $D_j$  thể hiện phương án điều chỉnh tăng công suất thêm  $D_j$  oát cho trạm điện thứ  $T_j$  ( $1 \leq T_j \leq M$ ;  $1 \leq D_j \leq 10^{18}$ ).



Gọi  $\Sigma_M$  và  $\Sigma_Q$  tương ứng là tổng của tất cả các giá trị  $M$  và  $Q$  trong tất cả  $\mathcal{T}$  trường hợp test. Dữ liệu bảo đảm  $1 \leq \Sigma_M, \Sigma_Q \leq 2 \times 10^5$ .

Các số trên cùng một dòng cách nhau bởi dấu cách.

## Kết quả

Ghi ra file văn bản THREE.OUT:

- Gồm  $\mathcal{T}$  nhóm dòng, mỗi nhóm dòng là kết quả của trường hợp test tương ứng có cấu trúc sau:
  - Dòng thứ nhất ghi ra một số nguyên dương là tổng công suất lớn nhất tìm được với hiện trạng ban đầu.
  - Dòng thứ  $j$  trong số  $Q$  dòng tiếp theo ghi ra tổng công suất lớn nhất tìm được với phương án điều chỉnh thứ  $j$ .

## Ví dụ

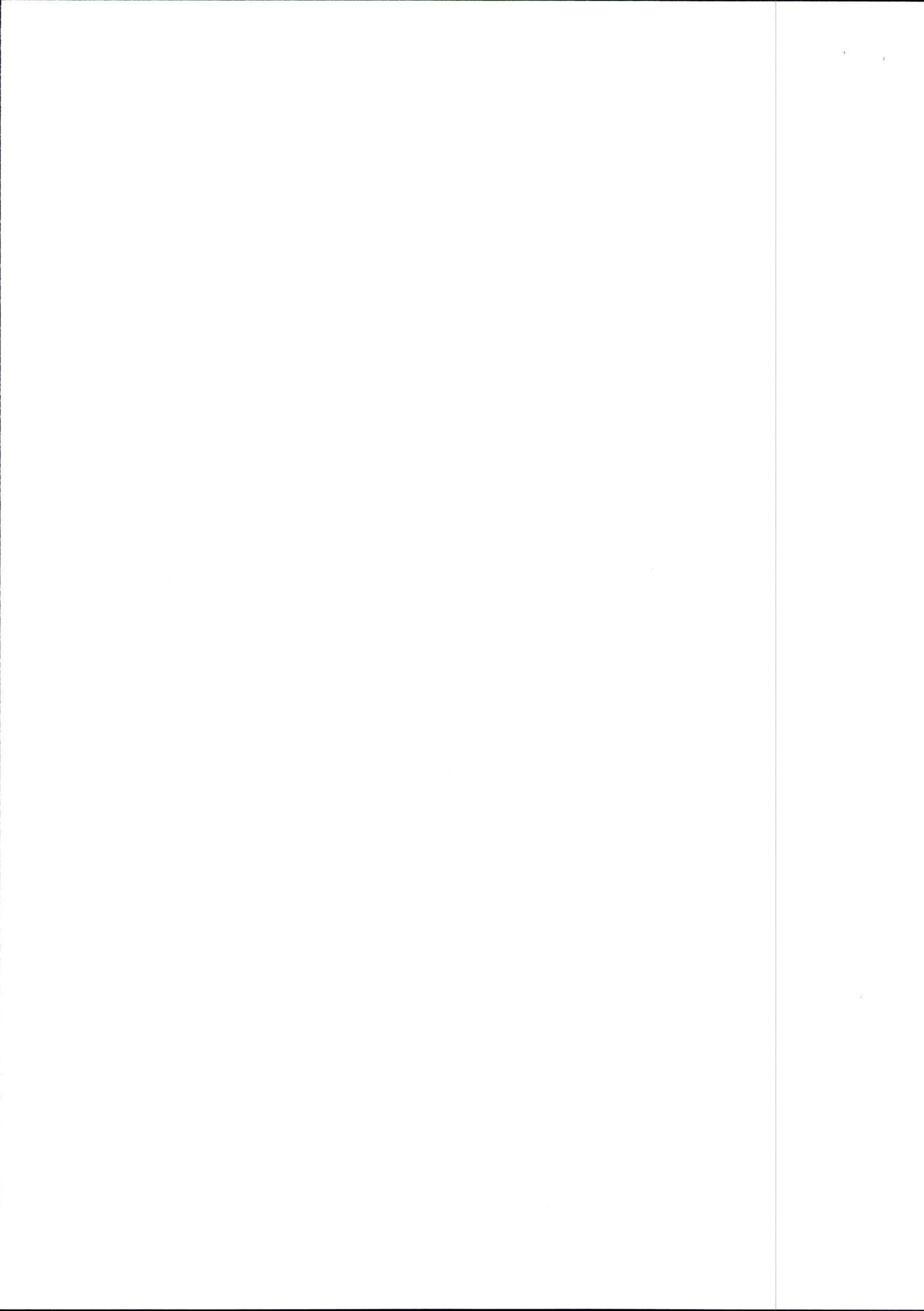
THREE.INP	THREE.OUT	Giải thích
2	17	Xét trường hợp test thứ nhất:
5 7 2	19	- Với hiện trạng ban đầu, một cách tối ưu là lắp ba đường truyền ở cột 1, cột 2 và cột 5. Tổng công suất cung cấp là 17.
1 1 1	24	- Với phương án điều chỉnh thứ nhất, một cách tối ưu là lắp ba đường truyền ở hàng 2, cột 2 và cột 5. Tổng công suất cung cấp là 19.
3 1 2	6	- Với phương án điều chỉnh thứ hai, một cách tối ưu là lắp ba đường truyền ở hàng 1, hàng 3 và cột 2. Tổng công suất cung cấp là 24.
2 2 3	7	
4 2 4		
2 4 2		
1 5 5		
3 5 2		
5 3		
2 7		
3 3 1		
1 1 1		
1 2 2		
1 3 3		
3 1		

	Cột 1	2	3	4	5
Hàng 1	1				5
2		3		2→5	
3	2				2
4		4			
5					

	Cột 1	2	3	4	5
Hàng 1	1				5
2		3		2	
3	2→9				2
4		4			
5					

## Ràng buộc

- (1) Có 20% số test ứng với 20% số điểm thỏa mãn:  $N, M, Q \leq 40$  và  $\mathcal{T} = 1$ .
- (2) 20% số test khác ứng với 20% số điểm thỏa mãn:  $N, M, Q \leq 100$  và  $\mathcal{T} = 1$ .
- (3) 20% số test khác ứng với 20% số điểm thỏa mãn:  $N, M, Q \leq 500$  và  $\mathcal{T} = 1$ .
- (4) 20% số test khác ứng với 20% số điểm thỏa mãn:  $M, Q \leq 1000$  và  $\Sigma_M, \Sigma_Q \leq 2000$ .
- (5) 10% số test khác ứng với 10% số điểm thỏa mãn:  $M \leq 1000$  và  $\Sigma_M \leq 2000$ .
- (6) 10% số test còn lại ứng với 10% số điểm: Không có ràng buộc gì thêm.



## Câu 2. Cải thiện đánh giá (7,0 điểm)

Quốc gia Beta có  $N$  thành phố được đánh số từ 1 đến  $N$ . Các thành phố được nối với nhau bởi  $M$  con đường hai chiều, được đánh số từ 1 đến  $M$ , giữa hai thành phố bất kỳ có tối đa một con đường nối trực tiếp chúng. Con đường hai chiều số  $i$  ( $1 \leq i \leq M$ ) nối trực tiếp giữa hai thành phố phân biệt  $U_i$  và  $V_i$  có độ dài  $W_i$ . Một đường đi gồm  $K$  thành phố từ thành phố  $X$  tới thành phố  $Y$  là một chuỗi các thành phố  $P_1, P_2, \dots, P_K$ , sao cho  $P_1 = X$ ,  $P_K = Y$  và có con đường trực tiếp giữa hai thành phố  $P_i$  và  $P_{i+1}$  ( $\forall i = 1, 2, \dots, K - 1$ ). Ở quốc gia Beta, mọi thành phố đều có đường đi tới thành phố khác. Chi phí di chuyển của một đường đi từ thành phố  $X$  tới thành phố  $Y$  là tổng độ dài của các con đường trên đường đi đó. Gọi  $D(X, Y)$  là chi phí di chuyển nhỏ nhất trong số các chi phí di chuyển của các đường đi từ thành phố  $X$  tới thành phố  $Y$ . Quy ước  $D(X, X) = 0$  với mọi thành phố  $X$ .

Thành phố số 1 có một nhà máy đúc khuôn và thành phố số 2 có một nhà máy sản xuất mạ tĩnh điện. Một doanh nghiệp muốn chọn một thành phố nào đó để mở một trung tâm chế tạo thép trang trí sử dụng nguyên liệu từ nhà máy đúc khuôn và nhà máy sản xuất mạ tĩnh điện. Thành phố  $Y$  được gọi là *tốt hơn hoặc tương đương* so với thành phố  $X$  khi và chỉ khi  $D(Y, 1) \leq D(X, 1)$  và  $D(Y, 2) \leq D(X, 2)$ . Lưu ý, thành phố  $X$  được xem là tốt hơn hoặc tương đương so với chính nó. Doanh nghiệp tính *hạng* của thành phố  $X$  là số lượng thành phố  $Y$  tốt hơn hoặc tương đương so với thành phố  $X$ . Cụ thể, công thức tính hạng là:

$$R(X) = |\{Y \in \{1, 2, \dots, N\} : D(Y, 1) \leq D(X, 1) \text{ và } D(Y, 2) \leq D(X, 2)\}|.$$

Ngoài ra, doanh nghiệp cũng cam kết với chính quyền địa phương rằng trước khi đặt trung tâm chế tạo thép ở thành phố  $X$  nào đó, họ sẽ cải tạo một con đường nối với thành phố  $X$ . Doanh nghiệp đã khảo sát và đưa ra  $P$  phương án. Với phương án thứ  $j$  ( $1 \leq j \leq P$ ), con đường  $T_j$  nối giữa thành phố  $U_{T_j}$  và  $V_{T_j}$  sẽ được chọn để cải tạo giúp cho độ dài mới  $W'_{T_j}$  bé hơn độ dài ban đầu  $W_{T_j}$ . Với mỗi phương án, sau khi cải tạo đường, doanh nghiệp cần tính hạng của thành phố  $U_{T_j}$  và thành phố  $V_{T_j}$ . Các phương án là độc lập, nghĩa là các phương án đều chỉ áp dụng lên hiện trạng ban đầu của  $M$  con đường.

**Yêu cầu:** Với phương án thứ  $j$  ( $1 \leq j \leq P$ ), hãy giúp doanh nghiệp tìm ra hạng của thành phố  $U_{T_j}$  và thành phố  $V_{T_j}$  sau khi cải tạo con đường  $T_j$ .

### Dữ liệu

Vào từ file văn bản IMPEVAL.INP:

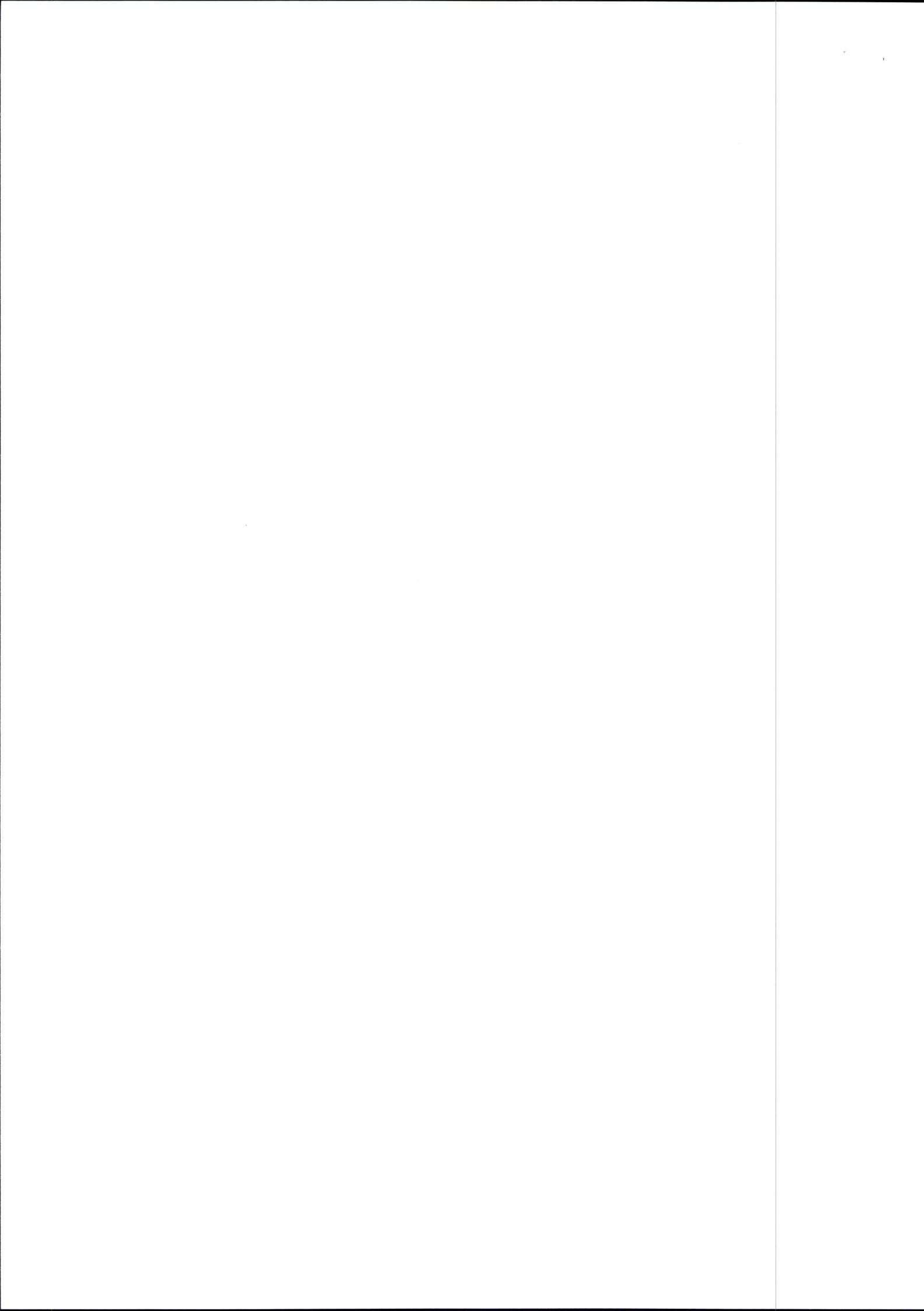
- Dòng đầu chứa ba số nguyên  $N, M$  và  $P$  lần lượt là số lượng thành phố, số lượng con đường và số lượng phương án ( $2 \leq N \leq 10^5; 1 \leq M, P \leq 10^5$ ).
- Dòng thứ  $i$  trong số  $M$  dòng tiếp theo chứa ba số nguyên  $U_i, V_i$  và  $W_i$  lần lượt là hai thành phố được nối bởi con đường số  $i$  và độ dài của con đường này ( $1 \leq U_i, V_i \leq N; 1 \leq W_i \leq 10^9$ ).
- Dòng thứ  $j$  trong số  $P$  dòng tiếp theo chứa hai số nguyên  $T_j$  và  $W'_{T_j}$  lần lượt là chỉ số con đường được lên phương án cải tạo và độ dài sau khi cải tạo ( $1 \leq T_j \leq M; 1 \leq W'_{T_j} < W_{T_j}$ ).

Các số trên cùng một dòng cách nhau bởi dấu cách.

### Kết quả

Ghi ra file văn bản IMPEVAL.OUT:

- Gồm  $P$  dòng, trong đó dòng thứ  $j$  ghi ra hai số nguyên  $R(U_{T_j})$  và  $R(V_{T_j})$  tương ứng là hạng của thành phố  $U_{T_j}$  và thành phố  $V_{T_j}$  nếu phương án thứ  $j$  được triển khai.

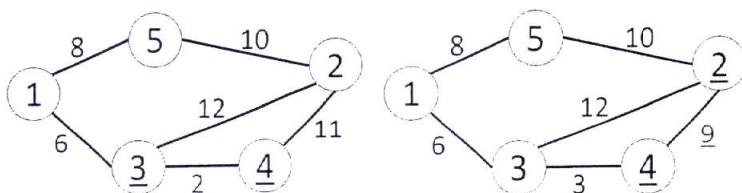


## Ràng buộc

- (1) Có 20% số test ứng với 20% số điểm thỏa mãn:  $M, P \leq 1000$ .
- (2) 20% số test khác ứng với 20% số điểm thỏa mãn: Mỗi thành phố có nhiều nhất 2 con đường nối với thành phố khác.
- (3) 20% số test khác ứng với 20% số điểm thỏa mãn: Mọi con đường đều nối với thành phố số 1 hoặc thành phố số 2.
- (4) 20% số test khác ứng với 20% số điểm thỏa mãn:  $M = N - 1$ .
- (5) 20% số test còn lại ứng với 20% số điểm: Không có ràng buộc gì thêm.

## Ví dụ

		Giải thích				
IMPEVAL.INP	IMPEVAL.OUT					
5 6 2	1 2	- Với con đường số 5 có độ dài mới bằng 2, thành phố số 3 có hạng 1 do chỉ có duy nhất thành phố số 3 tốt hơn hoặc tương đương với thành phố số 3, thành phố số 4 có hạng 2 do có hai thành phố tốt hơn hoặc tương đương với nó là thành phố số 4 và thành phố số 5.				
1 5 8	1 1	$\begin{array}{c ccccc} X & 1 & 2 & 3 & 4 & 5 \\ \hline D(X, 1) & 0 & 18 & 6 & 8 & 8 \\ \hline D(X, 2) & 18 & 0 & 12 & 11 & 10 \end{array}$				
5 2 10		- Với con đường số 6 có độ dài mới bằng 9, thành phố số 4 và thành phố số 2 đều có hạng 1.				
1 3 6						
3 2 12						
3 4 3						
4 2 11						
5 2						
6 9						

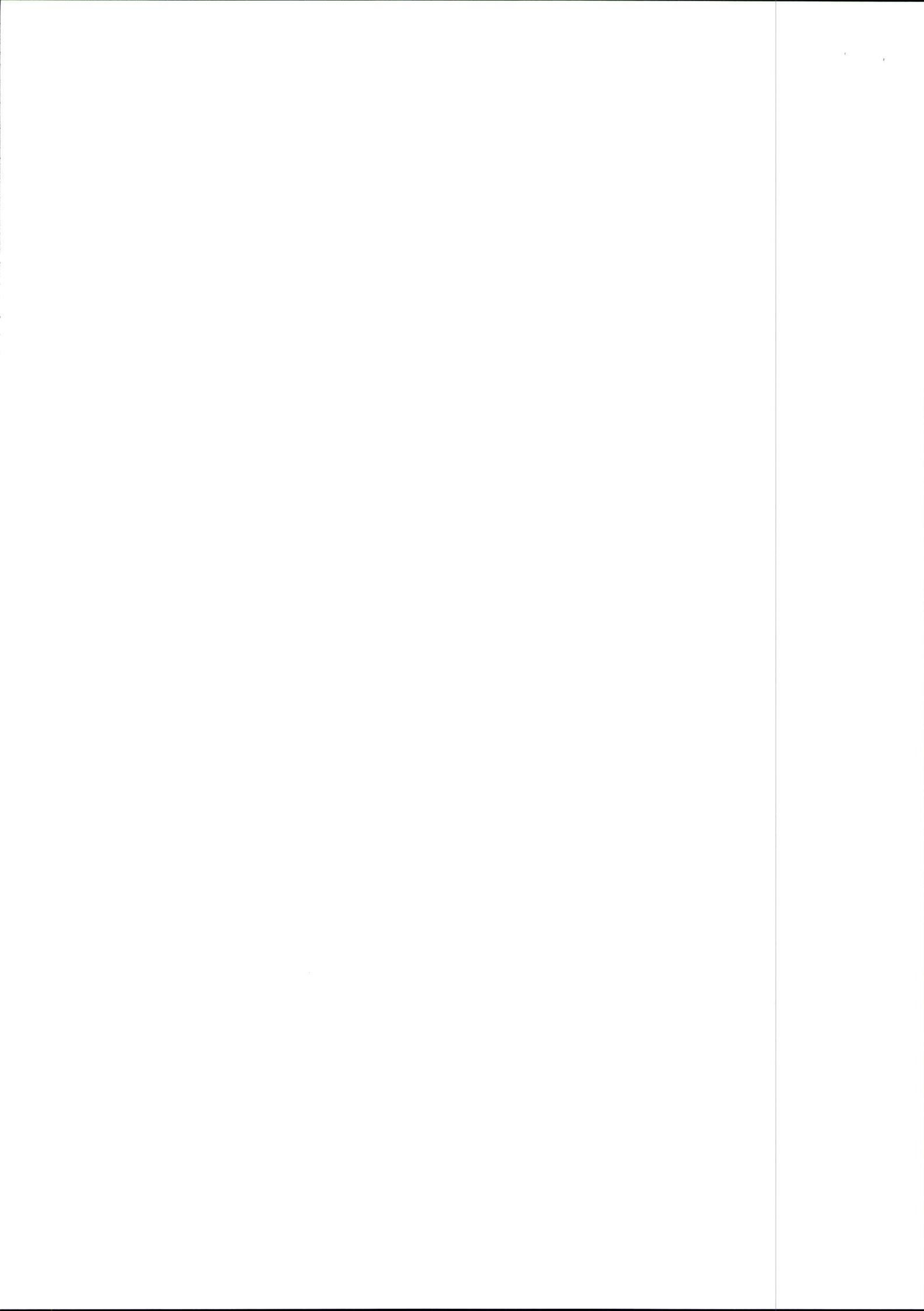


## Câu 3. Thu mua nông sản (6,0 điểm)

Quốc gia Delta có nền nông nghiệp hàng đầu thế giới. Năm nay, nhờ ứng dụng công nghệ thông tin sâu rộng trong sản xuất nông nghiệp, nông dân quốc gia Delta đã được mùa lớn. Để chúc mừng thành công lớn của bà con cũng như đẩy mạnh việc xuất khẩu, chính phủ quyết định bố trí các xe thu mua nông sản từ khắp mọi nơi trên cả nước.

Trong quốc gia có  $N$  ngôi làng được đánh số từ 1 đến  $N$ . Mạng lưới giao thông tại đây gồm  $N - 1$  con đường hai chiều, mỗi con đường nối trực tiếp hai ngôi làng nào đó. Các con đường này bảo đảm sự liên thông toàn quốc. Nói cách khác, từ một ngôi làng bất kỳ có thể đi tới tất cả các ngôi làng còn lại thông qua một hoặc nhiều con đường. Người dân tại quốc gia Delta có khả năng sản xuất được  $K$  loại nông sản khác nhau, được đánh số từ 1 đến  $K$ . Năm nay, người dân tại ngôi làng thứ  $i$  ( $1 \leq i \leq N$ ) sản xuất loại nông sản thứ  $A_i$ .

Chính phủ quốc gia Delta sẽ bố trí  $\frac{N \times (N-1)}{2}$  xe tải đi thu mua nông sản. Cụ thể, với mỗi cặp số nguyên  $(u, v)$  sao cho  $1 \leq u < v \leq N$ , có một xe tải xuất phát từ ngôi làng thứ  $u$ , đi qua một hoặc nhiều con đường rồi dừng lại ở ngôi làng thứ  $v$ . Biết rằng, xe tải luôn chọn đi theo tuyến đường qua ít con đường nhất có thể, và khi tới bất kỳ một ngôi làng nào (bao gồm cả hai ngôi làng thứ  $u$  và thứ  $v$ ), xe tải sẽ thu mua 1 tấn nông sản được sản xuất tại ngôi làng đó. Nhờ mùa màng bội thu, tất cả các ngôi làng đều có đủ nông sản cho mọi xe đi qua thu mua.



Việc vận chuyển nông sản luôn phát sinh chi phí. Tùy theo đặc tính, chi phí vận chuyển từng loại nông sản có thể khác nhau. Theo tính toán của chính phủ, nếu trong toàn bộ hành trình, một xe tải thu mua được khối lượng nông sản các loại thứ  $1, 2, \dots, K$  tương ứng là  $W_1, W_2, \dots, W_K$  tấn, chi phí vận chuyển của xe này là  $C_1 \times W_1^2 + C_2 \times W_2^2 + \dots + C_K \times W_K^2$ , trong đó  $C_1, C_2, \dots, C_K$  tương ứng là hệ số chi phí vận chuyển của  $K$  loại nông sản thứ  $1, 2, \dots, K$ . Chính phủ sẽ tài trợ toàn bộ chi phí vận chuyển, nên cần biết tổng chi phí của tất cả  $\frac{N \times (N-1)}{2}$  xe tải này.

Ngoài ra, với niềm tin rằng nền nông nghiệp còn phát triển mạnh trong nhiều năm về sau, chính phủ muốn dự trù chi phí vận chuyển nông sản cho những năm tiếp theo. Theo kế hoạch canh tác trong  $Q$  năm tiếp theo, vào năm thứ  $j$  ( $1 \leq j \leq Q$ ) ngôi làng thứ  $T_j$  sẽ chuyển qua sản xuất loại nông sản thứ  $B_j$ , trong khi  $N - 1$  ngôi làng còn lại sẽ tiếp tục canh tác loại nông sản như năm thứ  $j - 1$  (năm nay được coi là năm thứ 0). Với mỗi năm, chính phủ muốn biết tổng chi phí vận chuyển nông sản nếu tiếp tục bố trí các xe tải thu mua như phương án ở trên.

**Yêu cầu:** Hãy viết chương trình tính tổng chi phí vận chuyển nông sản của chính phủ trong năm nay và trong  $Q$  năm tiếp theo, dựa trên kế hoạch thay đổi canh tác.

## Dữ liệu

Vào từ file văn bản FBUY.INP:

- Dòng đầu chứa ba số nguyên  $N, K$  và  $Q$  lần lượt là số ngôi làng của quốc gia Delta, số loại nông sản được sản xuất tại đây và số năm trong kế hoạch canh tác ( $1 \leq K \leq 20; 1 \leq N, Q \leq 2 \times 10^5$ ).
- Dòng thứ hai chứa  $N$  số nguyên  $A_1, A_2, \dots, A_N$  thể hiện loại nông sản chuyên được sản xuất tại các ngôi làng trong năm nay ( $1 \leq A_i \leq K, \forall i = 1, 2, \dots, N$ ).
- Dòng thứ ba chứa  $K$  số nguyên  $C_1, C_2, \dots, C_K$  là hệ số chi phí vận chuyển của các loại nông sản ( $1 \leq C_i \leq 10^9, \forall i = 1, 2, \dots, K$ ).
- Mỗi dòng trong số  $N - 1$  dòng tiếp theo chứa hai số nguyên  $x$  và  $y$  cho biết có một con đường hai chiều nối ngôi làng thứ  $x$  và ngôi làng thứ  $y$ .
- Dòng thứ  $j$  trong số  $Q$  dòng cuối cùng chứa hai số nguyên  $T_j$  và  $B_j$  với ý nghĩa: Vào năm thứ  $j$  trong  $Q$  năm tiếp theo, ngôi làng thứ  $T_j$  sẽ chuyển sang sản xuất loại nông sản thứ  $B_j$  ( $1 \leq T_j \leq N; 1 \leq B_j \leq K$ ).

Các số trên cùng một dòng cách nhau bởi dấu cách.

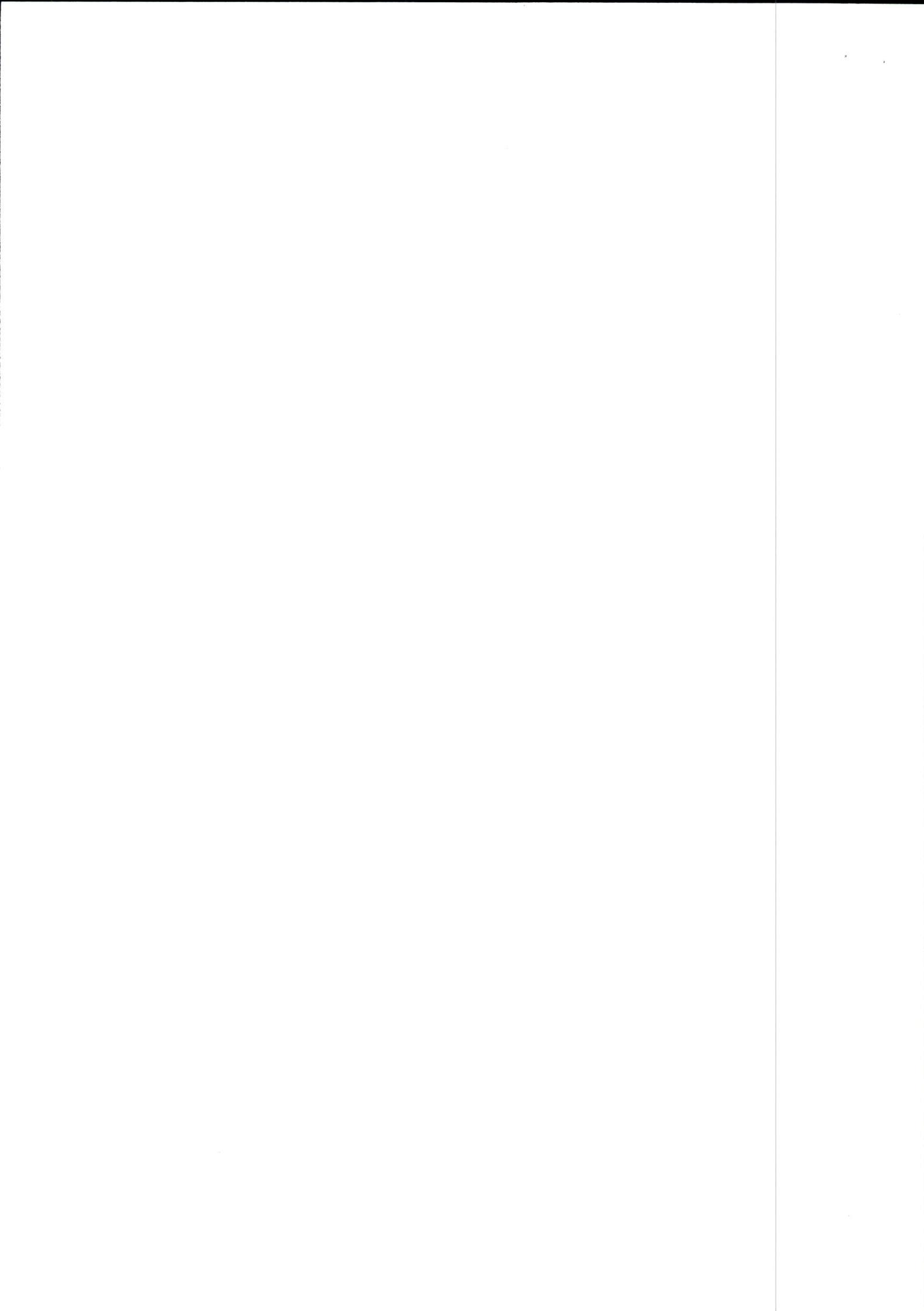
## Kết quả

Ghi ra file văn bản FBUY.OUT:

- Dòng đầu chứa một số nguyên là phần dư của tổng chi phí vận chuyển nông sản trong năm nay trong phép chia cho 998244353.
- Dòng thứ  $j$  trong số  $Q$  dòng còn lại chứa một số nguyên là phần dư của tổng chi phí vận chuyển nông sản trong năm thứ  $j$  trong phép chia cho 998244353.

## Ràng buộc

- (1) Có 7,5% số test ứng với 7,5% số điểm thỏa mãn:  $N \leq 30$  và  $Q \leq 800$ .
- (2) 12,5% số test khác ứng với 12,5% số điểm thỏa mãn:  $N \leq 100$  và  $Q \leq 800$ .
- (3) 10% số test khác ứng với 10% số điểm thỏa mãn:  $N \leq 2000$  và  $Q \leq 800$ .
- (4) 15% số test khác ứng với 15% số điểm thỏa mãn:  $N \leq 5000$  và  $Q \leq 8000$ .



- (5) 17,5% số test khác ứng với 17,5% số điểm thỏa mãn: Luôn tồn tại một con đường nối ngôi làng thứ  $i$  và ngôi làng thứ  $\lceil \frac{i}{2} \rceil$ ,  $\forall i = 2, 3, \dots, N$  ( $\lceil \frac{i}{2} \rceil$  là số nguyên lớn nhất không vượt quá  $\frac{i}{2}$ ).
- (6) 22,5% số test khác ứng với 22,5% số điểm thỏa mãn: Luôn tồn tại một con đường nối ngôi làng thứ  $i$  và ngôi làng thứ  $i - 1$ ,  $\forall i = 2, 3, \dots, N$ .
- (7) 15% số test còn lại ứng với 15% số điểm: Không có ràng buộc gì thêm.

### Ví dụ

FBUY.INP	FBUY.OUT	Minh họa mạng lưới giao thông
5 3 2	120	
1 1 1 2 3	137	
2 3 5	139	
1 2		
2 3		
2 4		
1 5		
2 3		
3 2		

```

graph TD
    1((1)) --- 2((2))
    1 --- 5((5))
    2 --- 3((3))
    2 --- 4((4))
  
```

### Giải thích

Trong năm nay, các ngôi làng thứ 1, 2, 3, 4, 5 lần lượt sản xuất các loại nông sản thứ 1, 1, 2, 3. Có tất cả  $\frac{5 \times (5-1)}{2} = 10$  xe tải với các lộ trình vận chuyển và chi phí như sau:

Các ngôi làng đi qua	Lượng nông sản thu mua theo tần (loại 1, loại 2, loại 3)	Chi phí vận chuyển
1 → 2	(2,0,0)	$2 \times 2^2 + 3 \times 0^2 + 5 \times 0^2 = 8$
1 → 2 → 3	(3,0,0)	$2 \times 3^2 + 3 \times 0^2 + 5 \times 0^2 = 18$
1 → 2 → 4	(2,1,0)	$2 \times 2^2 + 3 \times 1^2 + 5 \times 0^2 = 11$
1 → 5	(1,0,1)	$2 \times 1^2 + 3 \times 0^2 + 5 \times 1^2 = 7$
2 → 3	(2,0,0)	$2 \times 2^2 + 3 \times 0^2 + 5 \times 0^2 = 8$
2 → 4	(1,1,0)	$2 \times 1^2 + 3 \times 1^2 + 5 \times 0^2 = 5$
2 → 1 → 5	(2,0,1)	$2 \times 2^2 + 3 \times 0^2 + 5 \times 1^2 = 13$
3 → 2 → 4	(2,1,0)	$2 \times 2^2 + 3 \times 1^2 + 5 \times 0^2 = 11$
3 → 2 → 1 → 5	(3,0,1)	$2 \times 3^2 + 3 \times 0^2 + 5 \times 1^2 = 23$
4 → 2 → 1 → 5	(2,1,1)	$2 \times 2^2 + 3 \times 1^2 + 5 \times 1^2 = 16$

Tổng chi phí vận chuyển của các xe là  $8 + 18 + 11 + 7 + 8 + 5 + 13 + 11 + 23 + 16 = 120$ .

Theo kế hoạch canh tác các năm tiếp theo:

- Trong năm thứ 1, các ngôi làng thứ 1, 2, 3, 4, 5 sẽ lần lượt sản xuất các loại nông sản thứ 1, 3, 1, 2, 3. Số lượng xe và lộ trình di chuyển của các xe vẫn giống như ở trên, nhưng tổng chi phí vận chuyển của các xe là:  $7 + 13 + 10 + 7 + 7 + 8 + 22 + 10 + 28 + 25 = 137$ .
- Trong năm thứ 2, các ngôi làng thứ 1, 2, 3, 4, 5 sẽ lần lượt sản xuất các loại nông sản thứ 1, 3, 2, 2, 3. Số lượng xe và lộ trình di chuyển của các xe vẫn giống như ở trên, nhưng tổng chi phí vận chuyển của các xe là:  $7 + 10 + 10 + 7 + 8 + 8 + 22 + 17 + 25 + 25 = 139$ .

----- HẾT -----

\* Thí sinh KHÔNG được sử dụng tài liệu;

\* Giám thi KHÔNG được giải thích gì thêm.

1970-1971

1970-1971



KỲ THI CHỌN HỌC SINH GIỎI QUỐC GIA  
TRUNG HỌC PHỔ THÔNG  
NĂM HỌC 2023 - 2024

Môn: TIN HỌC

Thời gian: 180 phút (không kể thời gian giao đề)

Ngày thi thứ hai: 06/01/2024

(Đề thi gồm 06 trang, 03 câu)

TỔNG QUAN ĐỀ THI

	Tiêu đề	File chương trình	File dữ liệu	File kết quả
Câu 4	Sản xuất gỗ	WPRO.*	WPRO.INP	WPRO.OUT
Câu 5	Mạng truyền tin	NETW.*	NETW.INP	NETW.OUT
Câu 6	Bài tập đêm giáng sinh	NOEL.*	NOEL.INP	NOEL.OUT

Dấu \* được thay thế bởi PAS hoặc CPP tương ứng với ngôn ngữ lập trình Pascal hoặc C++.

Hãy lập trình giải các câu sau:

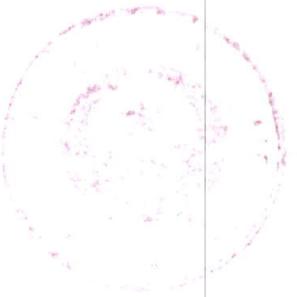
**Câu 4. Sản xuất gỗ** (7,0 điểm)

WoodPro là một nhà máy nổi tiếng chuyên sản xuất các sản phẩm về gỗ. Do nhu cầu thị trường ngày càng cao, nhà máy quyết định nhập khẩu một dây chuyền thông minh sản xuất ra hàng loạt những thanh gỗ. Mỗi một lượt sản xuất, dây chuyền sẽ nhận vào  $N$  thanh gỗ dạng hình trụ có cùng kích thước đáy. Các thanh gỗ được đánh số từ 1 đến  $N$ , thanh gỗ thứ  $i$  có độ dài  $A_i$  xăng-ti-mét. Thứ tự các thanh gỗ đưa vào dây chuyền là  $1, 2, \dots, N$ . Khi dây chuyền bắt đầu hoạt động, các thanh gỗ được xếp nối đuôi nhau trên một băng chuyền, theo đúng thứ tự nhận vào. Băng chuyền này sẽ di chuyển các thanh gỗ đi theo một công đoạn xử lý, trước tiên là *công đoạn cắt* rồi đến *công đoạn dán*, mà vẫn giữ nguyên thứ tự của các thanh gỗ trên băng chuyền.

- Ở công đoạn cắt, có một lưỡi cắt được đặt cố định phía trên băng chuyền. Mỗi khi có thanh gỗ di chuyển qua, hệ thống có thể điều khiển lưỡi cắt hạ xuống để cắt thanh gỗ thành hai thanh có tổng độ dài bằng độ dài của thanh gỗ trước khi cắt. Sau khi cắt, vị trí của hai thanh gỗ trên băng chuyền vẫn được giữ nguyên. Chi phí cho mỗi lần cắt như vậy là  $C$ .
- Ở công đoạn dán, có một máy dán được đặt cố định phía trên băng chuyền. Mỗi khi có hai thanh gỗ kề nhau di chuyển qua, hệ thống có thể điều khiển máy dán hạ xuống để dán hai thanh gỗ này thành một thanh gỗ có độ dài bằng tổng độ dài của hai thanh gỗ trước khi dán. Sau khi dán, vị trí của thanh gỗ trên băng chuyền vẫn được giữ nguyên. Chi phí cho mỗi lần dán như vậy là  $D$ .

Tuấn là một lập trình viên của nhà máy đảm nhận nhiệm vụ lập trình cho hệ thống đối với yêu cầu của mỗi đơn hàng. Do mới nhận được một đơn hàng yêu cầu các thanh gỗ thành phẩm chỉ gồm loại độ dài  $L_1$  xăng-ti-mét hoặc loại độ dài  $L_2$  xăng-ti-mét, nhà máy giao cho Tuấn lập trình cho hệ thống để sản xuất ra các thanh gỗ thành phẩm như vậy từ  $N$  thanh gỗ đầu vào mà không để thừa thanh gỗ nào có độ dài khác  $L_1$  và  $L_2$ .

**Yêu cầu:** Biết rằng luôn tồn tại một phương án sản xuất ra các thanh gỗ thành phẩm độ dài  $L_1$  và  $L_2$  từ  $N$  thanh gỗ đầu vào mà không thừa ra thanh gỗ nào có độ dài khác  $L_1$  và  $L_2$ , hãy tính tổng chi phí ít nhất có thể dùng cho việc cắt và dán, để dây chuyền sản xuất có thể hoàn thành được đơn hàng.



## Dữ liệu

Vào từ file văn bản WPRO.INP:

- Dòng đầu chứa năm số nguyên  $N, L_1, L_2, C$  và  $D$  lần lượt là số lượng thanh gỗ, hai loại độ dài các thanh gỗ thành phẩm đầu ra, chi phí cho một lần cắt và chi phí cho một lần dán ( $1 \leq N \leq 10^4; 1 \leq L_1, L_2 \leq 10^9; 1 \leq C, D \leq 10^5$ ).
- Dòng thứ hai chứa  $N$  số nguyên  $A_1, A_2, \dots, A_N$  là độ dài của  $N$  thanh gỗ đầu vào ( $1 \leq A_i \leq 10^9, \forall i = 1, 2, \dots, N$ ).

Dữ liệu bảo đảm luôn có phương án giải quyết theo yêu cầu đề bài.

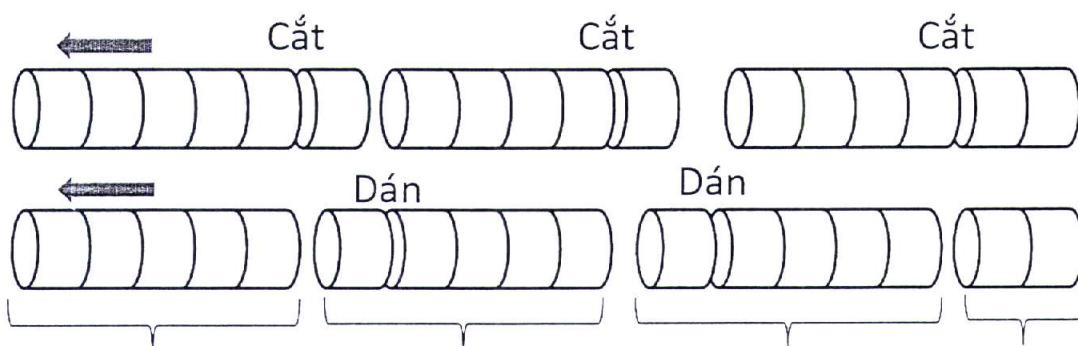
Các số trên cùng một dòng cách nhau bởi dấu cách.

## Kết quả

Ghi ra file văn bản WPRO.OUT một số nguyên là tổng chi phí dùng cho việc cắt và dán tìm được.

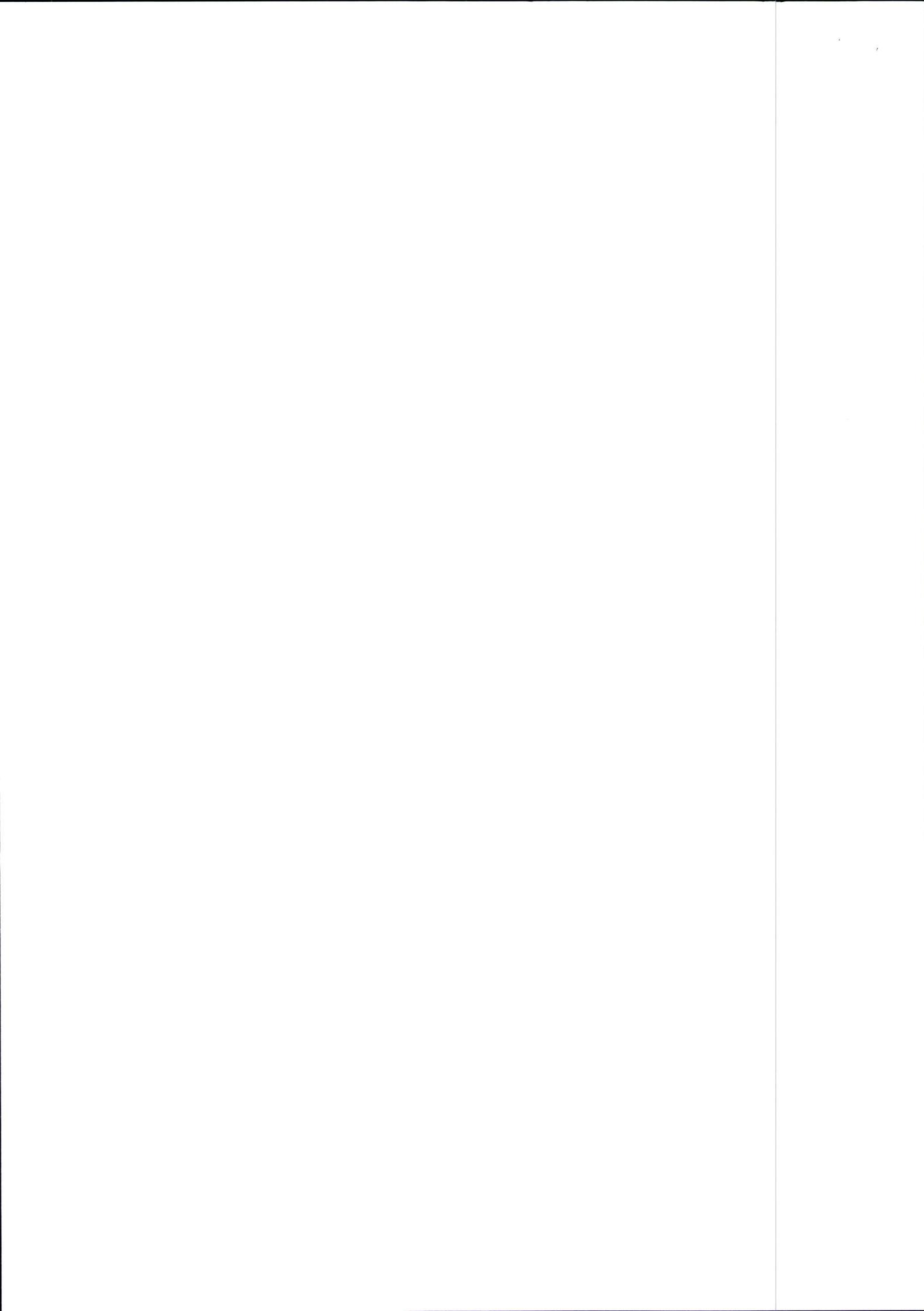
## Ví dụ

WPRO.INP	WPRO.OUT	Giải thích
3 2 5 10 4 6 5 6	38	- Một phương án tối ưu là dây chuyền thực hiện 3 lần cắt và thực hiện 2 lần dán như trong hình vẽ minh họa ở dưới. - Tổng chi phí là $10 + 10 + 10 + 4 + 4 = 38$ .
3 1 1 2 3 3 4 5	18	Phương án tối ưu là dây chuyền thực hiện 9 lần cắt.
3 12 13 2 3 3 4 5	6	Phương án tối ưu là dây chuyền thực hiện 2 lần dán.



## Ràng buộc

- (1) Có 16% số test ứng với 16% số điểm thỏa mãn:  $L_1 = L_2$ .
- (2) 16% số test khác ứng với 16% số điểm thỏa mãn:  $\sum_{i=1}^N A_i \leq 20$ .
- (3) 16% số test khác ứng với 16% số điểm thỏa mãn:  $N, L_1, L_2 \leq 100$  và  $A_i \leq 100, \forall i = 1, 2, \dots, N$ .
- (4) 16% số test khác ứng với 16% số điểm thỏa mãn:  $A_i \leq 2024, \forall i = 1, 2, \dots, N$ .
- (5) 12% số test khác ứng với 12% số điểm thỏa mãn:  $L_1, L_2 \leq 2024$ .
- (6) 12% số test khác ứng với 12% số điểm thỏa mãn:  $L_1 \leq 2024$ .
- (7) 12% số test còn lại ứng với 12% số điểm: Không có ràng buộc gì thêm.



## Câu 5. Mạng truyền tin (7,0 điểm)

Một mạng truyền tin có  $N$  máy tính, các máy tính được đánh số từ 1 đến  $N$ . Có  $N - 1$  dây cáp, được đánh số từ 1 đến  $N - 1$ . Dây cáp thứ  $i$  nối máy tính  $u_i$  với máy tính  $v_i$  và có giới hạn truyền tải  $w_i$ . Các dây cáp bảo đảm từ một máy tính bất kỳ có thể truyền tin đến tất cả các máy tính còn lại theo dây cáp trực tiếp giữa chúng hoặc thông qua đường truyền tin đi qua một số máy tính và dây cáp nào đó. Khi hai máy tính truyền tin cho nhau, chúng sẽ luôn sử dụng đường truyền tin sao cho không sử dụng dây cáp nào quá một lần. Độ lớn của gói tin truyền đi phải không lớn hơn giới hạn truyền tải của mọi dây cáp mà nó sử dụng. Chi phí để truyền một gói tin giữa hai máy tính bằng kích thước của gói tin nhân với bình phương số lượng dây cáp trên đường truyền tin.

Người ta muốn chọn ra một máy tính  $r$  để làm máy chủ. Khi đó, máy tính  $r$  sẽ truyền tin đến tất cả các máy tính khác. Vì phải dự trù cho mọi trường hợp, ta cần phải xét chi phí truyền tin tối đa. *Chi phí truyền tin tối đa* giữa máy tính  $r$  và máy tính  $x$  là  $C_{\min}(r, x) \times (D(r, x))^2$  với  $C_{\min}(r, x)$  là giới hạn truyền tải nhỏ nhất trong số các giới hạn truyền tải của các dây cáp trên đường truyền tin giữa máy tính  $r$  và máy tính  $x$ ,  $D(r, x)$  là số dây cáp trên đường truyền tin giữa máy tính  $r$  và máy tính  $x$ . *Chi phí vận hành mạng* là tổng của chi phí truyền tin tối đa giữa máy tính  $r$  và tất cả các máy tính khác.

**Yêu cầu:** Với mỗi  $r = 1, 2, \dots, N$ , hãy tính chi phí vận hành mạng nếu chọn máy tính  $r$  làm máy chủ.

### Dữ liệu

Vào từ file văn bản NETW.INP:

- Dòng đầu chứa một số nguyên  $N$  là số lượng máy tính ( $1 \leq N \leq 10^5$ ).
- Dòng thứ  $i$  trong số  $N - 1$  dòng tiếp theo chứa ba số nguyên  $u_i, v_i$  và  $w_i$  cho biết có một dây cáp nối máy tính  $u_i$  với máy tính  $v_i$  và có giới hạn truyền tải là  $w_i$  ( $1 \leq u_i, v_i \leq N$ ;  $1 \leq w_i \leq 10^9$ ).

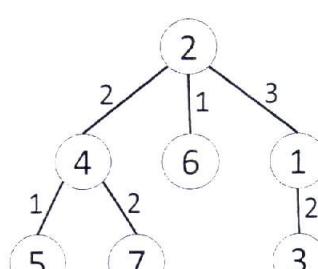
Các số trên cùng một dòng cách nhau bởi dấu cách.

### Kết quả

Ghi ra file văn bản NETW.OUT:

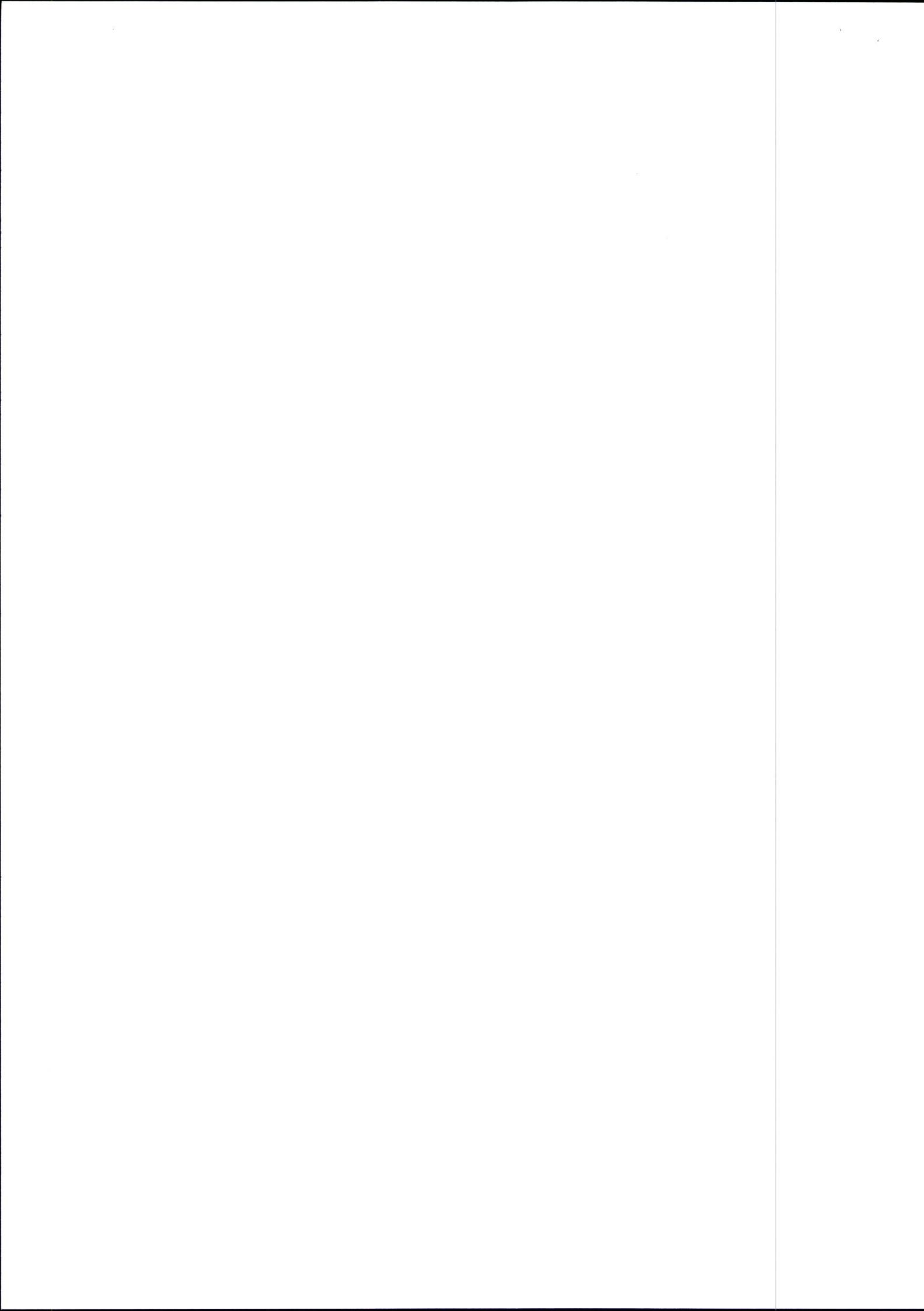
- Gồm  $N$  dòng, trong đó dòng thứ  $r$  chứa một số nguyên là phần dư của chi phí vận hành mạng nếu chọn máy tính  $r$  làm máy chủ trong phép chia cho 998244353.

### Ví dụ

NETW.INP	NETW.OUT	Minh họa mạng truyền tin
7	44	
1 2 3	26	
1 3 2	85	
2 4 2	35	
2 6 1	43	
4 5 1	36	
4 7 2	73	

### Giải thích

Chi phí truyền tin tối đa giữa tất cả các cặp máy tính được cho trong bảng sau:



$r \setminus x$	1	2	3	4	5	6	7	Tổng
1	-	3	2	8	9	4	18	44
2	3	-	8	2	4	1	8	26
3	2	8	-	18	16	9	32	85
4	8	2	18	-	1	4	2	35
5	9	4	16	1	-	9	4	43
6	4	1	9	4	9	-	9	36
7	18	8	32	2	4	9	-	73

Có 4 dây cáp trên đường truyền tin giữa máy tính 3 và máy tính 7 với giới hạn truyền tải là 2, 3, 2, 2, vì vậy  $C_{\min}(3, 7) = 2$  và  $D(3, 7) = 4$ . Chi phí truyền tin tối đa giữa 3 và 7 là  $2 \times 4^2 = 32$ , do đó số ở vị trí tương ứng với  $r = 3$  và  $x = 7$  trong bảng trên là 32.

Chi phí vận hành nếu chọn máy tính 4 làm máy chủ là  $8 + 2 + 18 + 1 + 4 + 2 = 35$ , do đó số ở cột cuối cùng ứng với  $r = 4$  trong bảng trên là 35.

### Ràng buộc

- (1) Có 16% số test ứng với 16% số điểm thỏa mãn:  $N \leq 5000$ .
- (2) 12% số test khác ứng với 12% số điểm thỏa mãn:  $w_i \leq 2$  và luôn có dây cáp nối giữa máy tính  $i$  và máy tính  $i + 1$ ,  $\forall i = 1, 2, \dots, N - 1$ .
- (3) 20% số test khác ứng với 20% số điểm thỏa mãn:  $w_i = 1$ ,  $\forall i = 1, 2, \dots, N - 1$ .
- (4) 16% số test khác ứng với 16% số điểm thỏa mãn:  $w_i \leq 1000$ ,  $\forall i = 1, 2, \dots, N - 1$ .
- (5) 16% số test khác ứng với 16% số điểm thỏa mãn: Luôn có dây cáp nối giữa máy tính  $i$  và máy tính  $i + 1$ ,  $\forall i = 1, 2, \dots, N - 1$ .
- (6) 20% số test còn lại ứng với 20% số điểm: Không có ràng buộc gì thêm.

### Câu 6. Bài tập đêm giáng sinh (6,0 điểm)

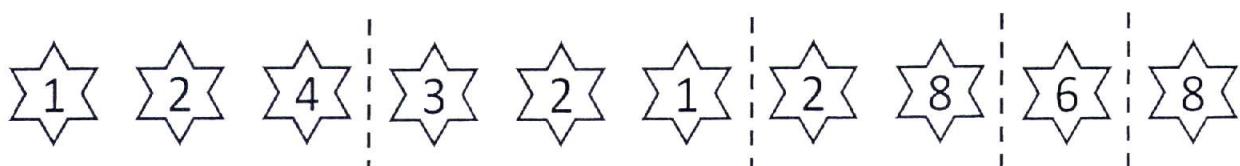
Cô Tuyết chuẩn bị một bài tập đặc biệt dành cho các bạn trong đội tuyển học sinh giỏi vào giáng sinh năm nay. Đó là một bài tập về thứ tự từ điển với đề bài như sau.

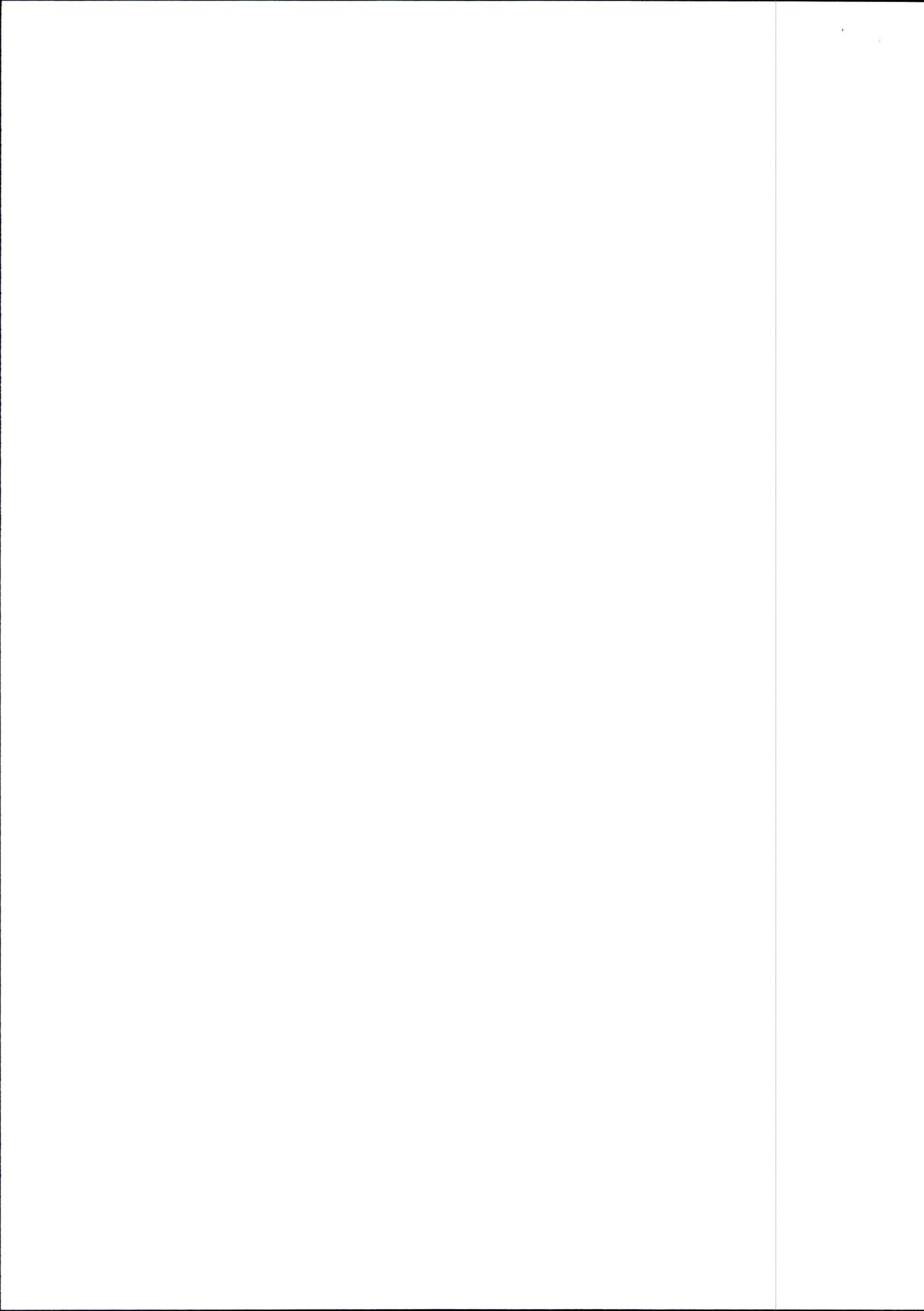
Cho một dãy số khác rỗng  $C = [C_1, C_2, \dots, C_M]$  thỏa mãn  $C_i \neq C_{i-1}$ ,  $\forall i = 2, 3, \dots, M$ . Ta gọi một cách *phân đoạn dãy* là một cách chia dãy thành các đoạn con chứa các phần tử liên tiếp, mà mỗi phần tử đều thuộc đúng một đoạn con. Một cách phân đoạn dãy được coi là *hợp lệ* nếu mỗi đoạn con chỉ chứa các phần tử đôi một phân biệt.

Ví dụ, với  $M = 10$  và dãy  $C = [1, 2, 4, 3, 2, 1, 2, 8, 6, 8]$ , thì một cách phân đoạn dãy hợp lệ là chia dãy  $C$  thành 4 đoạn con lần lượt là  $[1, 2, 4, 3], [2, 1], [2, 8], [6, 8]$ .



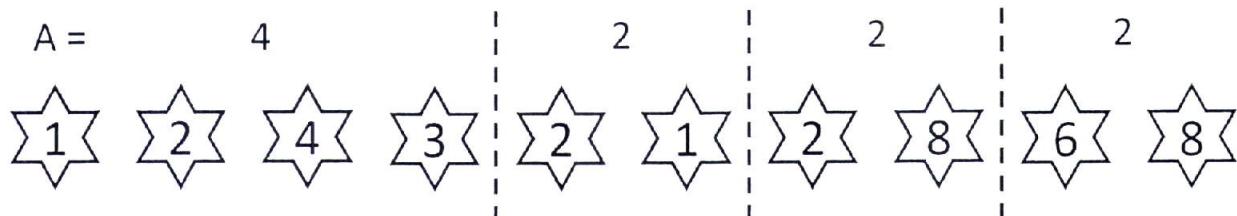
Một cách phân đoạn hợp lệ khác là chia dãy  $C$  thành 5 đoạn con lần lượt là  $[1, 2, 4], [3, 2, 1], [2, 8], [6], [8]$ .



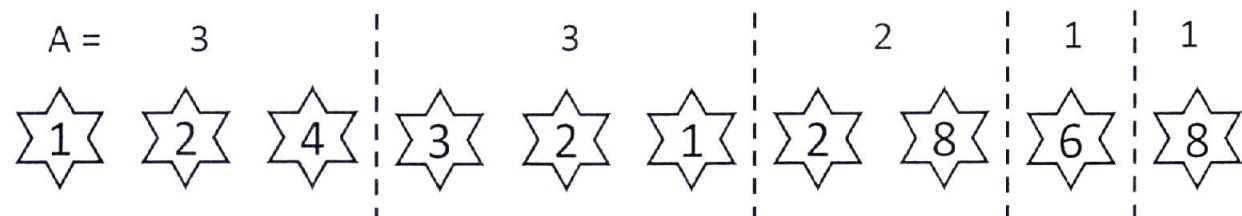


Cả hai cách phân đoạn dãy trên đều thỏa mãn mọi đoạn con chỉ chứa các phần tử đôi một phân biệt. Mặt khác, nếu ta chia dãy  $C$  thành 4 đoạn con lần lượt là  $[1, 2, 4, 3], [2, 1], [2], [8, 6, 8]$  thì sẽ không hợp lệ bởi có hai phần tử cùng bằng 8 trong đoạn con cuối cùng. Tương tự, nếu ta chia dãy  $C$  thành 5 đoạn con lần lượt là  $[1], [2, 4, 3, 2, 1, 2], [8], [6], [8]$  thì cũng không hợp lệ bởi có ba phần tử cùng bằng 2 trong đoạn con thứ hai.

Mỗi dãy có thể có nhiều cách phân đoạn dãy hợp lệ. Mỗi cách phân đoạn dãy được mã hóa bằng một *dãy mã hóa phân đoạn*  $A = [A_1, A_2, \dots, A_K]$ , với  $K$  là số lượng đoạn con, và  $A_i$  là số lượng phần tử của đoạn con thứ  $i$ . Chẳng hạn, với  $M = 10$  và dãy  $C = [1, 2, 4, 3, 2, 1, 2, 8, 6, 8]$ , thì cách phân đoạn dãy trong hình đầu tiên sẽ được mã hóa bằng dãy  $A = [4, 2, 2, 2]$ .



Tương tự, cách phân đoạn dãy trong hình thứ hai sẽ được mã hóa thành dãy  $A = [3, 3, 2, 1, 1]$ .



Cô Tuyết viết lên bảng một dãy số khác rỗng  $C = [C_1, C_2, \dots, C_M]$  thỏa mãn  $C_i \neq C_{i-1}$ ,  $\forall i = 2, 3, \dots, M$ , rồi lại viết lên bảng thêm 2 số nguyên dương  $X$  và  $Y$ . Sau đó cô gọi một bạn lên bảng để trả lời câu hỏi sau:

“Khi liệt kê tất cả các dãy mã hóa phân đoạn của dãy  $C$  rồi sắp xếp chúng theo thứ tự từ điển ngược thì dãy mã hóa phân đoạn thứ  $X$  và dãy mã hóa phân đoạn thứ  $Y$  có độ dài tiền tố chung dài nhất là bao nhiêu?”

**Nhắc lại:** Một dãy mã hóa phân đoạn  $U = [U_1, U_2, \dots, U_K]$  gọi là đi trước dãy mã hóa phân đoạn  $V = [V_1, V_2, \dots, V_H]$  theo *thứ tự từ điển ngược* nếu thỏa mãn một trong hai điều kiện sau:

- $K > H$  và  $U_i = V_i, \forall i = 1, 2, \dots, H$ ;
- Tồn tại  $1 \leq T \leq \min(K, H)$  sao cho  $U_i = V_i, \forall i = 1, 2, \dots, T - 1$  và  $U_T > V_T$ .

Dãy số  $U = [U_1, U_2, \dots, U_K]$  được gọi là *tiền tố* độ dài  $K$  của dãy  $V = [V_1, V_2, \dots, V_H]$  khi và chỉ khi  $K \leq H$  và  $U_i = V_i, \forall i = 1, 2, \dots, K$ . Lưu ý, dãy rỗng (ký hiệu là  $[]$ ) là tiền tố độ dài 0 của mọi dãy số.

*Tiền tố chung dài nhất* của hai dãy số  $Z_1$  và  $Z_2$  là dãy số có độ dài lớn nhất vừa là tiền tố của dãy  $Z_1$  vừa là tiền tố của dãy  $Z_2$ .

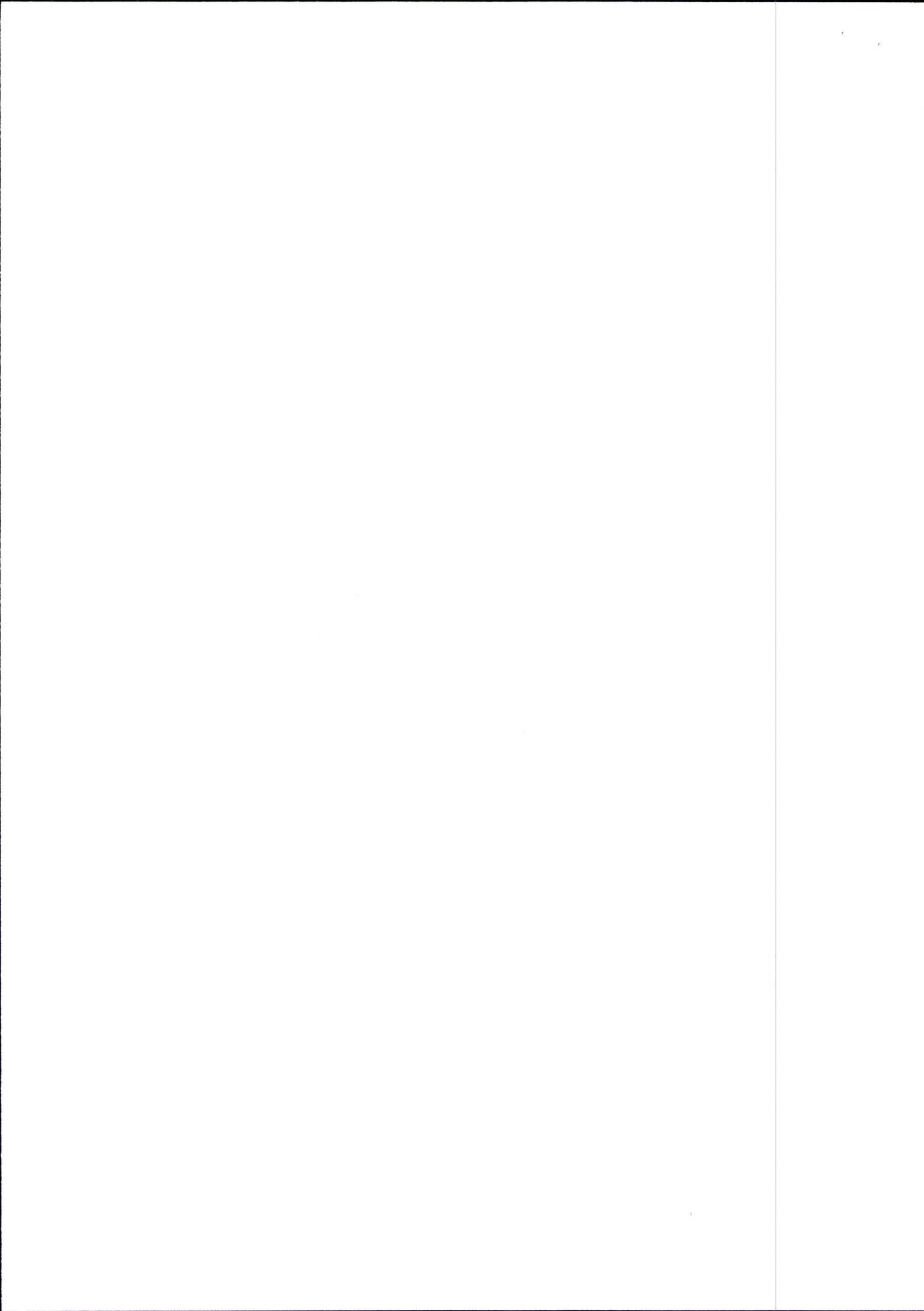
Nhận thấy rằng nếu chỉ đặt ra câu hỏi cho một cặp số  $(X, Y)$  thì bài toán chưa đủ khó, cô Tuyết quyết định thêm vào một vài tham số mới. Cô gọi  $Q$  bạn lần lượt lên bảng, bạn thứ  $i$  sẽ nhận được 4 số  $L_i, R_i, X_i, Y_i$  và có nhiệm vụ trả lời câu hỏi sau:

“Khi liệt kê tất cả các dãy mã hóa phân đoạn của dãy  $C' = [C_{L_i}, C_{L_i+1}, \dots, C_{R_i}]$  rồi sắp xếp chúng theo thứ tự từ điển ngược thì dãy mã hóa phân đoạn thứ  $X_i$  và dãy mã hóa phân đoạn thứ  $Y_i$  có độ dài tiền tố chung dài nhất là bao nhiêu?”

**Yêu cầu:** Hãy giúp các bạn trả lời  $Q$  câu hỏi của cô Tuyết.

## Dữ liệu

Vào từ file văn bản NOEL.INP:



- Dòng đầu chứa hai số nguyên  $M$  và  $Q$  lần lượt là độ dài của dãy  $C$  và số lượng bạn được cõi Tuyết gọi lên bảng ( $1 \leq M, Q \leq 2 \times 10^5$ ).
- Dòng thứ hai chứa  $M$  số nguyên  $C_1, C_2, \dots, C_M$  ( $1 \leq C_i \leq M, \forall i = 1, 2, \dots, M; C_i \neq C_{i-1}, \forall i = 2, 3, \dots, M$ ).
- Dòng thứ  $i$  trong số  $Q$  dòng tiếp theo biểu diễn một câu hỏi với bốn số nguyên  $L_i, R_i, X_i, Y_i$  ( $1 \leq L_i \leq R_i \leq M; 1 \leq X_i, Y_i \leq 10^{17}$ ). Dữ liệu bảo đảm số lượng dãy mã hóa phân đoạn của dãy  $C' = [C_{L_i}, C_{L_i+1}, \dots, C_{R_i}]$  không nhỏ hơn  $\max(X_i, Y_i)$ .

Các số trên cùng một dòng cách nhau bởi dấu cách.

### Kết quả

Ghi ra file văn bản NOEL.OUT:

- Gồm  $Q$  dòng, trong đó dòng thứ  $i$  chứa một số nguyên là độ dài tiền tố chung dài nhất của hai dãy mã hóa phân đoạn của dãy  $C' = [C_{L_i}, C_{L_i+1}, \dots, C_{R_i}]$ , với dãy mã hóa thứ nhất có thứ tự từ điển ngược bằng  $X_i$  và dãy mã hóa thứ hai có thứ tự từ điển ngược bằng  $Y_i$ .

### Ví dụ

NOEL.INP	NOEL.OUT	Giải thích
10 2	2	- Trong câu hỏi đầu tiên, dãy cần xét đến là $C' = [2, 4, 3, 2]$ . Tất cả các dãy mã hóa phân đoạn sắp xếp theo thứ tự từ điển ngược lần lượt là $[3, 1]; [2, 2]; [2, 1, 1]; [1, 3]; [1, 2, 1]; [1, 1, 2]; [1, 1, 1, 1]$ . Hai dãy mã hóa phân đoạn có thứ tự từ điển ngược bằng 6 và bằng 7 tương ứng là $Z_1 = [1, 1, 2]$ và $Z_2 = [1, 1, 1, 1]$ . Độ dài tiền tố chung dài nhất của hai dãy $Z_1$ và $Z_2$ là 2.
1 2 4 3 2 1 2 8 6 8	0	- Trong câu hỏi thứ hai, dãy cần xét đến là $C' = [3, 2, 1, 2, 8, 6, 8]$ . Hai dãy mã hóa phân đoạn có thứ tự từ điển ngược bằng 7 và bằng 6 tương ứng là $Z_1 = [2, 4, 1]$ và $Z_2 = [3, 1, 1, 1, 1]$ . Độ dài tiền tố chung dài nhất của hai dãy $Z_1$ và $Z_2$ là 0.
2 5 6 7		
4 10 7 6		

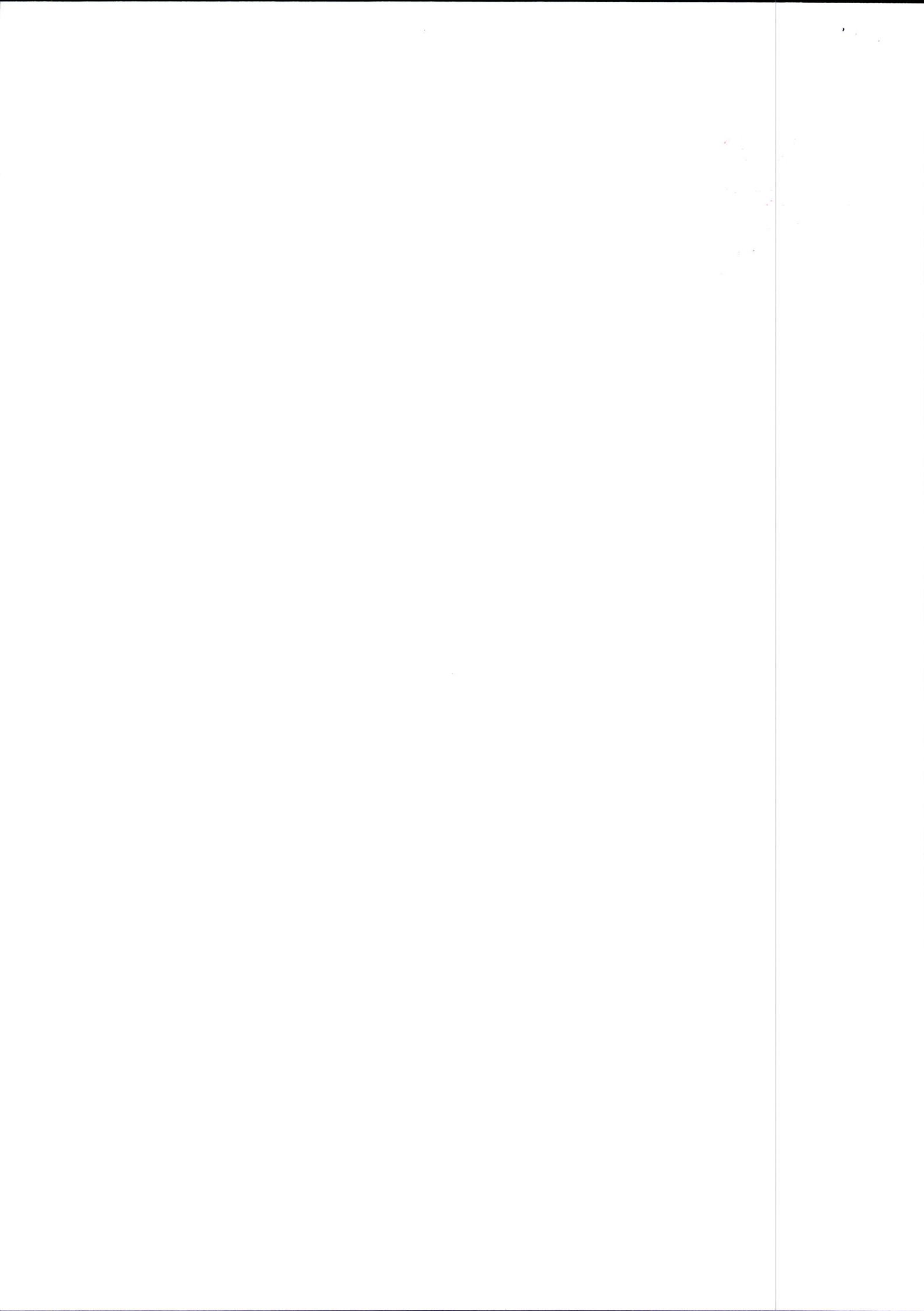
### Ràng buộc

- (1) Có 17,5% số test ứng với 17,5% số điểm thỏa mãn:  $M \leq 12$  và  $Q \leq 1024$ .
- (2) 20% số test khác ứng với 20% số điểm thỏa mãn:  $X_i = Y_i = 1, \forall i = 1, 2, \dots, Q$ .
- (3) 22,5% số test khác ứng với 22,5% số điểm thỏa mãn:  $L_i = 1; R_i = M$  và  $X_i, Y_i \leq 10^5, \forall i = 1, 2, \dots, Q$ .
- (4) 22,5% số test khác ứng với 22,5% số điểm thỏa mãn:  $M \leq 2000$  và  $Q \leq 20000$ .
- (5) 17,5% số test còn lại ứng với 17,5% số điểm: Không có ràng buộc gì thêm.

----- HẾT -----

\* Thí sinh KHÔNG được sử dụng tài liệu;

\* Giám thi KHÔNG được giải thích gì thêm.





KỲ THI CHỌN HỌC SINH GIỎI QUỐC GIA  
TRUNG HỌC PHỔ THÔNG  
NĂM HỌC 2023 - 2024

Môn: TIN HỌC

TỔNG QUAN ĐỀ THI

	Tiêu đề	File chương trình	File dữ liệu	File kết quả
Câu 1	Ba đường truyền điện	THREE.*	THREE.INP	THREE.OUT
Câu 2	Cải thiện đánh giá	IMPEVAL.*	IMPEVAL.INP	IMPEVAL.OUT
Câu 3	Thu mua nông sản	FBUY.*	FBUY.INP	FBUY.OUT
Câu 4	Sản xuất gỗ	WPRO.*	WPRO.INP	WPRO.OUT
Câu 5	Mạng truyền tin	NETW.*	NETW.INP	NETW.OUT
Câu 6	Bài tập đêm giáng sinh	NOEL.*	NOEL.INP	NOEL.OUT

Dấu \* được thay thế bởi PAS hoặc CPP tương ứng với ngôn ngữ lập trình Pascal hoặc C++.

**Câu 1. Ba đường truyền điện (7,0 điểm)**

- Subtask 1: Có 20% số test ứng với 20% số điểm thỏa mãn:  $N, M, Q \leq 40$  và  $\mathcal{T} = 1$ .

Duyệt tất cả các trường hợp: 3 đường dọc, 3 đường ngang, 2 dọc 1 ngang và 1 ngang 2 dọc. Số lượng 3 đường có thể chọn là  $O(N^3)$ , việc tính toán cho mỗi đường là  $O(N)$ . Với mỗi truy vấn, cập nhật giá trị điểm và tính lại.

Độ phức tạp tính toán của thuật toán là  $O(N^4 \times Q)$ .

- Subtask 2: 20% số test khác ứng với 20% số điểm thỏa mãn:  $N, M, Q \leq 100$  và  $\mathcal{T} = 1$ .

Sử dụng việc tính trước tổng các hàng và các cột làm cho việc tính tổng 3 đường tính cả việc cập nhật là  $O(1)$ . Với mỗi truy vấn, độ phức tạp tính là  $O(N^3)$ .

Độ phức tạp tính toán của thuật toán là  $O(N^3 \times Q)$ .

- Subtask 3: 20% số test khác ứng với 20% số điểm thỏa mãn:  $N, M, Q \leq 500$  và  $\mathcal{T} = 1$ .

Ta có nhận xét rằng việc chọn 3 đường ngang hoặc 3 đường dọc tốt nhất chỉ mất độ phức tạp thuật toán là  $O(N)$ . Còn với việc có 1 đường ngang (dọc) và 2 đường dọc (ngang), ta duyệt theo hàng ngang và cập nhật giá trị của 2 đường dọc tốt nhất. Độ phức tạp thuật toán cho mỗi truy vấn là  $O(N^2)$ .

Độ phức tạp tính toán của thuật toán là  $O(N^2 \times Q)$ .

- Subtask 4: 20% số test khác ứng với 20% số điểm thỏa mãn:  $M, Q \leq 1000$  và  $\Sigma_M, \Sigma_Q \leq 2000$ .

Cải tiến từ subtask 3, ta sử dụng cấu trúc dữ liệu để cập nhật và truy vấn 2 đường dọc tốt nhất sau khi cập nhật đường ngang. Thêm vào đó, ta sử dụng việc rời rạc hóa bài toán để số lượng đường cần duyệt là  $O(M)$ . Độ phức tạp thuật toán cho mỗi truy vấn là  $O(M \log M)$ .

Độ phức tạp tính toán của thuật toán là  $O(M \log M \times Q)$ .

- Subtask 5: 10% số test khác ứng với 10% số điểm thỏa mãn:  $M \leq 1000$  và  $\Sigma_M \leq 2000$ .

Nhận xét là kết quả bài toán chỉ thay đổi khi 1 trong 3 đường đi qua điểm mới cập nhật. Nên ta sẽ tính trước tất cả kết quả tối ưu cho 2 đường còn lại. Khi đó, phương án tối ưu nếu buộc sử dụng điểm mới cập nhật là công suất tính trước cộng thêm công suất tăng thêm. Để tính trước 2 đường còn lại, ta duyệt 1 đường và tìm ra đường tối ưu còn lại bằng cấu trúc dữ liệu.



Số lượng trường hợp cần tính trước là  $O(M^2)$ . Độ phức tạp tính toán cho việc tính toán trước là  $O(M^2 \log M)$ .

Độ phức tạp tính toán của thuật toán là  $O(M^2 \log M + Q)$ .

- **Subtask 6:** 10% số test còn lại ứng với 10% số điểm:  $N \leq 10^9; M, Q \leq 10^5; \Sigma_M, \Sigma_Q \leq 2 \times 10^5$ .

Tương tự Subtask 5, ta duyệt 1 đường và tìm 2 đường tốt nhất còn lại. Nhưng với mỗi đường dọc ta chỉ cần lưu lại 2 đường đi có tổng lớn nhất mà không quan tâm đến tính dọc ngang của nó. Ta chỉ cần lưu trữ  $O(M)$  phương án, sử dụng cấu trúc dữ liệu việc tìm phương án mất  $O(\log M)$ .

Độ phức tạp tính toán của thuật toán là  $O(M \log M + Q)$ .

## Câu 2. Cải thiện đánh giá (7,0 điểm)

- **Subtask 1:** Có 20% số test ứng với 20% số điểm thỏa mãn:  $M, P \leq 1000$ .

Với mỗi truy vấn, ta thực hiện thuật toán tìm đường đi ngắn nhất Dijkstra hai lần với đỉnh 1 và đỉnh 2. Độ phức tạp tính toán là  $O((N + M) \log N)$ . Sau đó, ta đếm từng đỉnh  $z$  thỏa mãn điều kiện  $D(1, z) \leq D(1, x)$  và  $D(2, z) \leq D(2, x)$ . Ta được thứ hạng của  $x$ , làm tương tự với  $y$ . Độ phức tạp của bước này là  $O(N)$ .

Độ phức tạp chung của thuật toán là  $O(Q \times M \times \log N)$  do  $N \leq M + 1$ .

- **Subtask 2:** Có 20% số test khác ứng với 20% số điểm thỏa mãn: Mỗi thành phố có nhiều nhất 2 con đường nối với thành phố khác.

Do đồ thị là liên thông nên chỉ có 2 trường hợp: đồ thị là một đường thẳng hoặc đồ thị là một chu trình duy nhất.

– Trong trường hợp đồ thị là một đường thẳng:

\* Nếu  $u$  nằm trên đường đi từ 1 tới 2 thì hạng của  $u$  là 1.

\* Nếu  $u$  không nằm trên đường đi từ 1 tới 2:

· Nếu  $D(u, 1) \leq D(u, 2)$ : ta dùng phương pháp tìm kiếm nhị phân với kết quả là số đỉnh gần 1 hơn so với  $u$ .

· Nếu  $D(u, 1) > D(u, 2)$ : ta dùng phương pháp tìm kiếm nhị phân với kết quả là số đỉnh gần 2 hơn so với  $u$ .

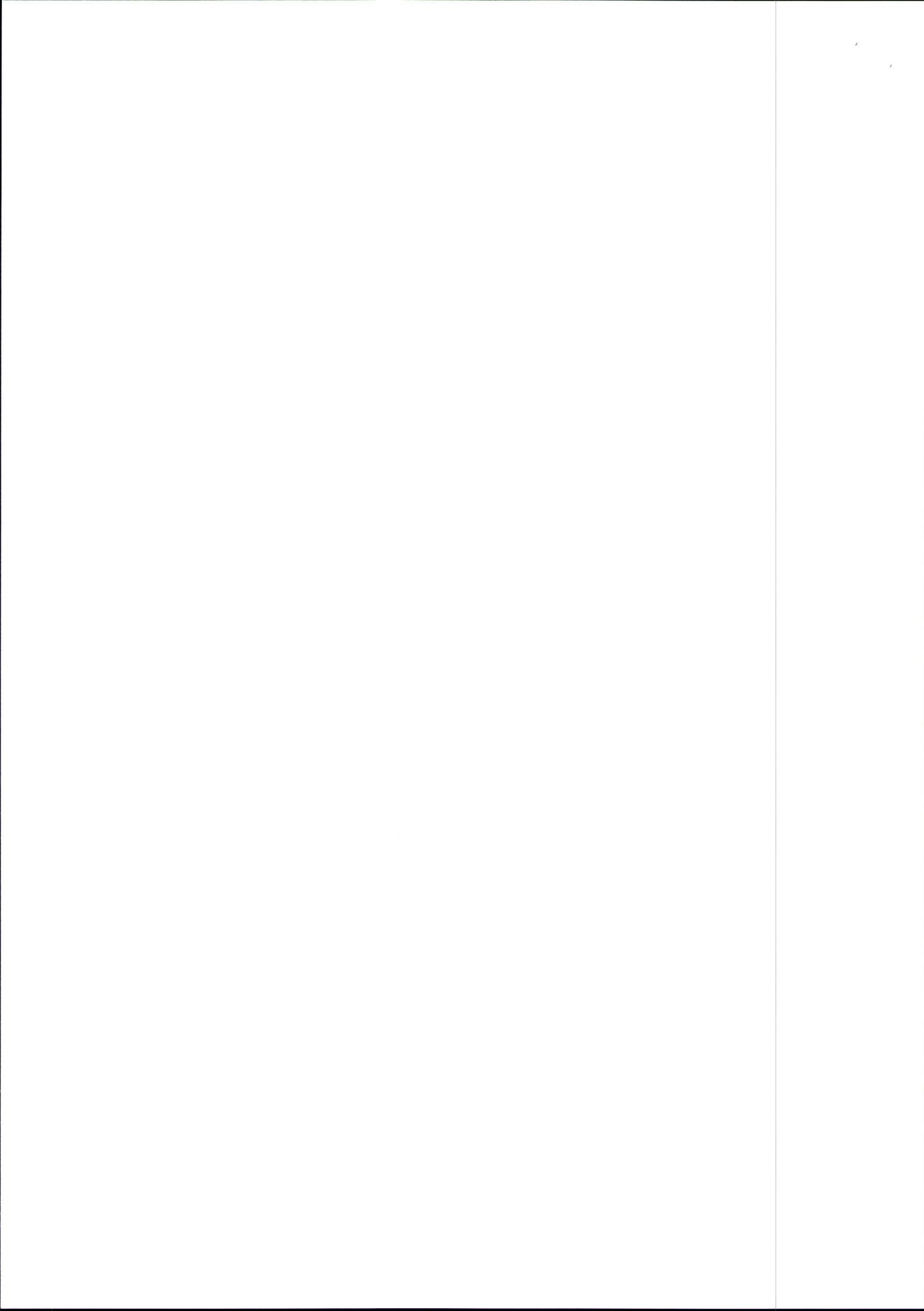
– Trong trường hợp đồ thị là một chu trình: Sẽ có 2 đường đi khác nhau từ 1 tới 2. Ta áp dụng phương pháp tìm kiếm nhị phân trên 2 đường này để tính số đỉnh tốt hơn hoặc tương đương  $U$  và  $V$ .

Độ phức tạp tính toán là  $O(Q \times \log N)$ .

- **Subtask 3:** Có 20% số test khác ứng với 20% số điểm thỏa mãn: Mọi con đường có một trong hai đầu mút là thành phố số 1 hoặc thành phố số 2.

Với đồ thị dạng này, cạnh được cập nhật chi phí mới sẽ không làm thay đổi chi phí của các đỉnh khác. Với mỗi truy vấn, ta cập nhật giá trị  $D'(u, 1)$  và  $D'(u, 2)$ . Bài toán quy về cho  $N$  điểm  $(x, y)$  trên mặt phẳng, mỗi truy vấn cần đếm xem có bao nhiêu điểm  $(x, y)$  thỏa mãn  $x \leq D(u, 1)$  và  $y \leq D(u, 2)$ . Bài toán này ta có thể xử lý offline bằng cách sweep line theo tọa độ  $x$  và dùng một cấu trúc dữ liệu dạng cây để đếm xem có bao nhiêu điểm có  $y \leq D(u, 2)$  trong độ phức tạp  $O(\log N)$ .

Độ phức tạp chung của thuật toán là  $O(Q \times \log N)$ .



- **Subtask 4:** Có 20% số test khác ứng với 20% số điểm thỏa mãn: Quốc gia Beta có đúng  $N - 1$  con đường.

Ta cần nhận xét việc thay đổi một cạnh trên cây sẽ làm thay đổi chi phí của đỉnh khác nhưng những đỉnh bị thay đổi chi phí không ảnh hưởng tới thứ hạng của hai đỉnh truy vấn do chi phí của các con đường là dương. Sử dụng kỹ thuật xử lý offline như trên, ta có độ phức tạp chung của thuật toán là  $O(Q \times \log N)$ .

- **Subtask 5:** 20% số test còn lại ứng với 20% số điểm:  $1 \leq N, M, P \leq 10^5$ .

Ta cần nhận xét việc thay đổi một cạnh đồ thị tổng quát cũng có tính chất tương tự như trường hợp cây. Sử dụng kỹ thuật xử lý offline như trên, ta có độ phức tạp chung của thuật toán là  $O(Q \times \log N)$ .

### Câu 3. Thu mua nông sản (6,0 điểm)

- **Subtask 1:** Có 7,5% số test ứng với 7,5% số điểm thỏa mãn:  $N \leq 30$  và  $Q \leq 800$ .

Mô phỏng lại toàn bộ các xe vận tải bằng kỹ thuật DFS.

Độ phức tạp là  $O(Q \times N^3)$ .

- **Subtask 2:** 12,5% số test khác ứng với 12,5% số điểm thỏa mãn:  $N \leq 100$  và  $Q \leq 800$ .

Thí sinh tính tổng chi phí bằng cách thực hiện  $n$  lần DFS từ  $n$  ngôi làng. Để tính chi phí cho cặp  $(u, v)$  ta DFS từ đỉnh  $u$  rồi tính chi phí cặp  $(u, v)$  dựa trên cặp  $(u, p_v)$  với  $p_v$  là cha của  $u$  trong  $O(1)$ .

Độ phức tạp là  $O(Q \times N^2)$ .

- **Subtask 3:** 10% số test khác ứng với 10% số điểm thỏa mãn:  $N \leq 2000$  và  $Q \leq 800$ .

Với năm thứ  $j$  trong  $Q$  năm tiếp theo, cần tính chênh lệch chi phí giữa năm thứ  $j$  và năm thứ  $j - 1$ . Để làm điều này, xét các ngôi làng cùng loại nông sản với ngôi làng thứ  $T_j$  rồi sử dụng kỹ thuật tìm tổ tiên chung gần nhất (LCA).

Độ phức tạp là  $O(Q \times N \times \log N)$ .

- **Subtask 4:** 15% số test khác ứng với 15% số điểm thỏa mãn:  $N \leq 5000$  và  $Q \leq 8000$ .

Thí sinh chuẩn bị trước mảng  $\text{cnt}[u][v]$  là số cặp đỉnh  $(x, y)$  sao cho đường đi giữa  $x$  và  $y$  chứa cả hai đỉnh  $u$  và  $v$ . Sau đó phần tính tổng chi phí thí sinh thực hiện như Subtask 3.

Độ phức tạp là  $O(Q \times N + N^2)$ .

- **Subtask 5:** 17,5% số test khác ứng với 17,5% số điểm thỏa mãn: Với mọi  $2 \leq i \leq N$ , luôn tồn tại một con đường nối ngôi làng thứ  $i$  và ngôi làng thứ  $\lceil \frac{i}{2} \rceil$ .

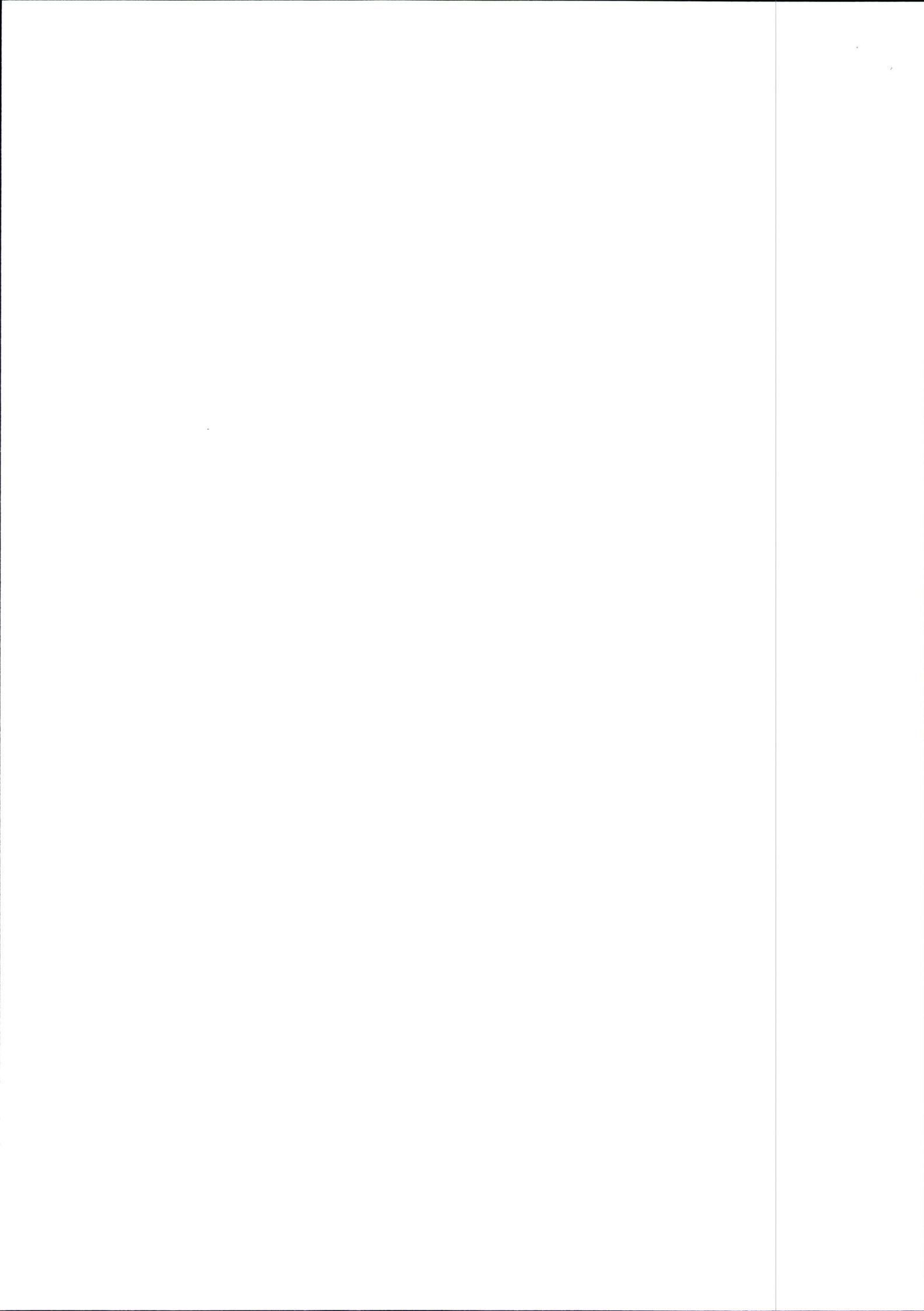
Nhận thấy rằng, đường đi giữa hai ngôi làng bất kỳ có không quá  $\log N + 1$  con đường. Do đó, thí sinh có thể tính chênh lệch chi phí giữa năm thứ  $j$  và năm thứ  $j - 1$  bằng cách duyệt qua mọi tổ tiên của ngôi làng  $T_j$ , sau đó dùng cấu trúc dữ liệu cây nhị phân chỉ số (BIT).

Độ phức tạp là  $O(Q \times \log^2 N)$ .

- **Subtask 6:** 22,5% số test khác ứng với 22,5% số điểm thỏa mãn: Với mọi  $2 \leq i \leq N$ , luôn tồn tại một con đường nối ngôi làng thứ  $i$  và ngôi làng thứ  $i - 1$ .

Nhận thấy rằng, với hai ngôi làng  $i$  và  $j$  bất kỳ ( $1 \leq i \leq j \leq n$ ), số cặp ngôi làng  $(x, y)$  sao cho đường đi từ  $x$  tới  $y$  chứa cả hai đỉnh  $i$  và  $j$  là  $i \times (n - j + 1)$ . Lợi dụng điều này, ta có thể tính chênh lệch chi phí giữa năm thứ  $j$  và năm thứ  $j - 1$  bằng cấu trúc dữ liệu cây phân đoạn (IT).

Độ phức tạp là  $O(Q \times \log N)$ .



- **Subtask 7:** 15% số test còn lại ứng với 15% số điểm thỏa mãn:  $1 \leq K \leq 20; 1 \leq N, Q \leq 2 \times 10^5$ .  
Sử dụng ý tưởng tương tự như các Subtask 3 - 6 nhưng phải kết hợp với kỹ thuật phân cảnh nặng nhẹ (Heavy-Light Decomposition).  
Độ phức tạp là  $O(Q \times \log^2 N)$ .

### Lời giải chi tiết:

Mạng lưới giao thông có cấu trúc dạng cây, trong đó mỗi ngôi là là một đỉnh. Sau đây ta sẽ gọi “đỉnh” thay cho “ngôi làng”.

Ta có  $W^2 = W + 2 \times \binom{W}{2}$  với mọi số nguyên dương  $W$ . Do đó, chi phí vận tải của xe tải đi từ đỉnh  $u$  tới đỉnh  $v$  có thể hiểu như sau: Xét các đỉnh trên đường đi từ đỉnh  $u$  tới đỉnh  $v$ :

- Mỗi đỉnh  $x$  sản xuất loại nông sản thứ  $i$  làm chi phí vận tải tăng thêm  $C_i$ .
- Mỗi cặp đỉnh  $(x, y)$  cùng sản xuất loại nông sản thứ  $i$  làm chi phí vận tải tăng thêm  $C_i$ .

Ta ký hiệu:

- $F_1(u)$  là số cặp đỉnh  $(x, y)$  sao cho đường đi từ  $x$  đến  $y$  chứa đỉnh  $u$ .
- $F_2(u, v)$  là số cặp đỉnh  $(x, y)$  sao cho đường đi từ  $x$  đến  $y$  chứa cả hai đỉnh  $u$  và  $v$ .

Khi đó, tổng chi phí của tất cả  $\frac{N \times (N-1)}{2}$  xe tải có thể hiểu như sau:

- Mỗi đỉnh  $x$  sản xuất loại nông sản thứ  $i$  làm chi phí vận tải tăng thêm  $C_i \times F_1(x)$ .
- Mỗi cặp đỉnh  $(x, y)$  cùng sản xuất loại nông sản thứ  $i$  làm chi phí vận tải tăng thêm  $C_i \times F_2(x, y)$ .

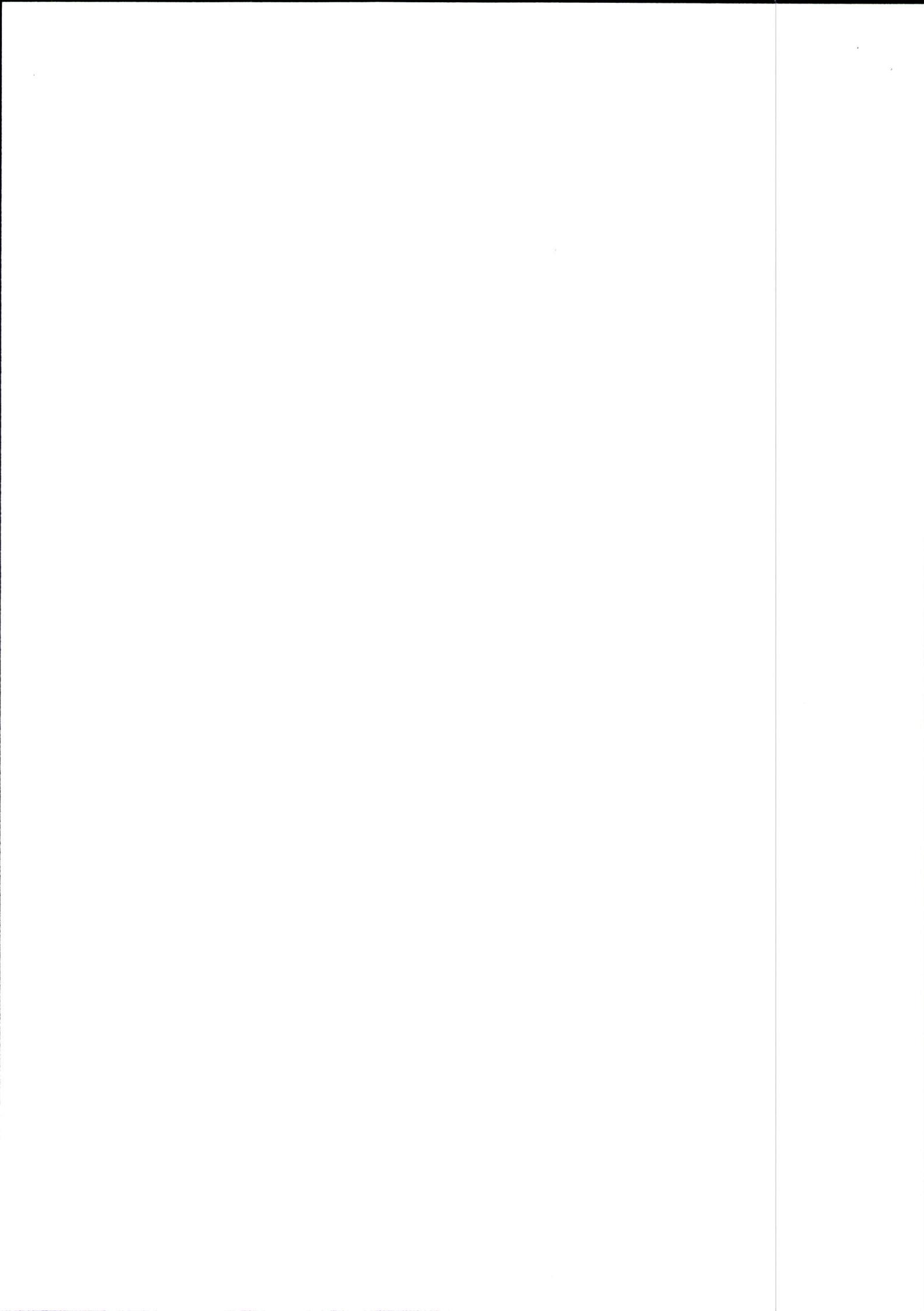
Ta tính giá trị  $F_1(u)$  với mọi đỉnh  $u$  bằng phương pháp quy hoạch động trên cây. Coi đỉnh 1 là gốc cây, gọi  $S(u)$  là số đỉnh trong cây con gốc  $u$ ; giá trị  $F_2(u, v)$  được xác định như sau:

- Nếu  $u$  là tổ tiên của  $v$ , gọi  $p$  là con trực tiếp của  $u$  sao cho  $v$  thuộc cây con gốc  $p$ , khi đó  $F_2(u, v) = S(v) \times (N - S(p))$ .
- Nếu  $v$  là tổ tiên của  $u$ , ta làm tương tự như trên.
- Nếu  $u$  và  $v$  không có đỉnh nào là tổ tiên của đỉnh còn lại,  $F_2(u, v) = S(u) \times S(v)$ .

Tới đây, việc tính lại tổng chi phí khi một đỉnh  $T_j$  chuyển loại nông sản từ  $b$  sang  $b'$ , ta chia làm hai quá trình độc lập:

- Tổng chi phí giảm đi do mất đi đỉnh  $T_j$  sản xuất nông sản loại  $b$ .
- Tổng chi phí tăng lên do có thêm đỉnh  $T_j$  sản xuất nông sản loại  $b'$ .

Việc tính lượng chi phí tăng lên hay giảm đi hoàn toàn như nhau, ta quy về bài toán sau: Cho đỉnh  $u$  và loại  $i$ , tính tổng  $F_2(u, v)$  với mọi đỉnh  $v$  sản xuất loại nông sản  $i$ . Với mỗi loại nông sản, ta dùng một cây nhị phân chỉ số quản lý các đỉnh có loại nông sản này. Khi đó, bài toán trên có thể được giải bằng kỹ thuật Heavy-Light Decomposition.



## Câu 4. Sản xuất gỗ (7,0 điểm)

- Subtask 1: Có 16% số test ứng với 16% số điểm thỏa mãn:  $L_1 = L_2$ .

Do chỉ có một phương án cắt dán duy nhất, ta chỉ cần tính chi phí cắt dán cho phương án đó.  
Độ phức tạp:  $O(N)$ .

- Subtask 2: 16% số test khác ứng với 16% số điểm thỏa mãn:  $\sum_{i=1}^N A_i \leq 20$ .

Duyệt vét cạn tất cả các cách cắt/dán có thể có.

Độ phức tạp của thuật toán là  $O(2^S \times N)$  với  $S = \sum_{i=1}^N A_i$ .

- Subtask 3: 16% số test khác ứng với 16% số điểm thỏa mãn:  $N, L_1, L_2, A_i \leq 100, \forall i = 1, 2, \dots, N$ .

Xét từng vị trí trên từng thanh gỗ, mỗi vị trí có thể giữ nguyên hoặc thực hiện việc cắt/dán. Do đó, cần duy trì được độ dài của thanh gỗ hiện tại. Sử dụng thuật toán quy hoạch động, với trạng thái bài toán là  $(i, j)$  cho biết vị trí trên băng chuyền là  $i$  và thanh gỗ hiện tại có độ dài  $j$ , chuyển nhãn băng cách xem xét có cắt/dán tại vị trí hiện tại hay không.

Độ phức tạp của thuật toán là  $O(S \times \max(L_1, L_2))$  với  $S = \sum_{i=1}^N A_i$ .

- Subtask 4: 16% số test khác ứng với 16% số điểm thỏa mãn:  $A_i \leq 2024, \forall i = 1, 2, \dots, N$ .

Nhận xét là độ dài của thanh gỗ thành phẩm chỉ có thể là  $L_1$  hoặc  $L_2$ , nên ta có thể tối ưu thuật toán quy hoạch động ở Subtask 2. Trạng thái bài toán là vị trí  $i$  trên băng chuyền, chuyển nhãn băng cách xem xét thanh gỗ tiếp theo có độ dài  $L_1$  hay  $L_2$ .

Độ phức tạp của thuật toán là  $O(S)$  với  $S = \sum_{i=1}^N A_i$ .

- Subtask 5: 12% số test khác ứng với 12% số điểm thỏa mãn:  $L_1, L_2 \leq 2024$ .

Nhận xét là có thể thực hiện thao tác dán trước sau đó mới cắt. Gọi  $F(i)$  là chi phí để cắt dán  $i$  thanh gỗ đầu tiên. Việc chuyển nhãn có thể thực hiện bằng cách xét đoạn liên tiếp sẽ dán vào với thanh gỗ thứ  $i$ . Giả sử cần dán các thanh gỗ từ  $j$  đến  $i$  lại với nhau, khi đó  $F(i) = F(j-1) + D \times (i-j) + C \times g(A_j + A_{j+1} + \dots + A_i)$  với  $g(x)$  là số lần cắt ít nhất để thanh gỗ có độ dài  $x$  trở thành các thanh gỗ có độ dài  $L_1$  và  $L_2$ . Do  $L_1$  và  $L_2$  bé nên có thể tính hàm  $g$  bằng cách chuẩn bị tất cả các trường hợp  $x \leq L_1 \times L_2$ .

Độ phức tạp của thuật toán là  $O(N^2 + L_1 \times L_2)$ .

- Subtask 6: 12% số test khác ứng với 12% số điểm thỏa mãn:  $L_1 \leq 2024$ .

Làm tương tự như Subtask 5 nhưng do  $L_1$  bé nên có thể tính hàm  $g$  bằng thuật toán tìm kiếm theo chiều rộng (BFS).

Độ phức tạp của thuật toán là  $O(N^2 + L_1^2)$ .

- Subtask 7: 12% số test còn lại ứng với 12% số điểm:  $1 \leq N \leq 10^4; 1 \leq L_1, L_2 \leq 10^9; 1 \leq C, D \leq 10^5$ .

Làm tương tự như Subtask 5 nhưng sử dụng thuật toán Euclidean mở rộng, giải phương trình nghiệm nguyên để tính hàm  $g$ .

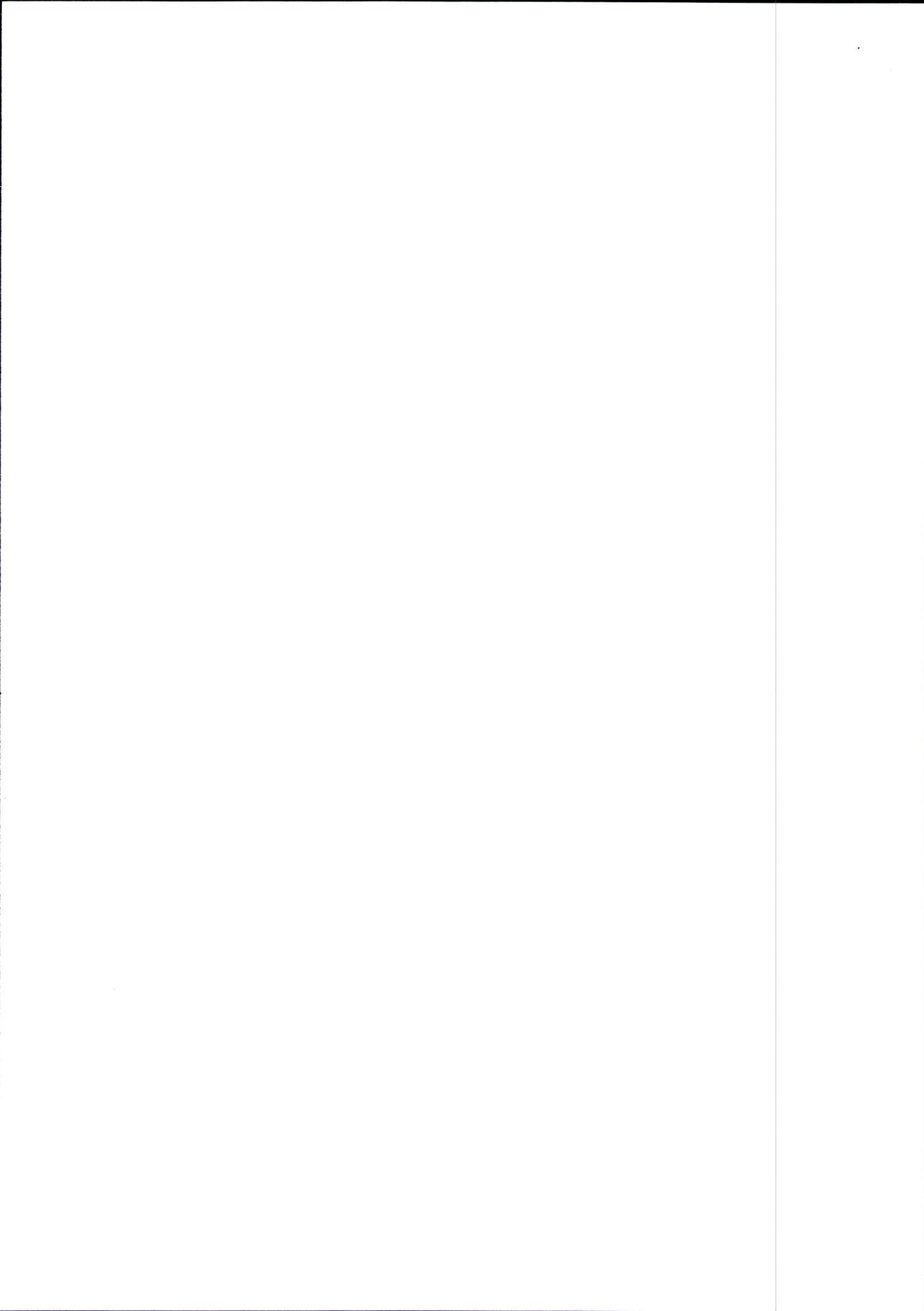
Độ phức tạp của thuật toán là  $O(N^2 + \log(L_1))$ .

## Câu 5. Mạng truyền tin (7,0 điểm)

- Subtask 1: Có 16% số test ứng với 16% số điểm thỏa mãn:  $N \leq 5000$ .

Tìm kiếm theo chiều sâu (DFS) lần lượt từ từng đỉnh.

Độ phức tạp tính toán:  $O(N^2)$



- **Subtask 2:** 12% số test khác ứng với 12% số điểm thỏa mãn:  $w_i \leq 2$  và luôn có dây cáp nối giữa máy tính  $i$  và máy tính  $i + 1$ ,  $\forall i = 1, 2, \dots, N - 1$ .

Xét hai trường hợp: Giới hạn truyền tải nhỏ nhất bằng hai hoặc bằng một. Với mỗi trường hợp, đưa về bài toán đếm (và tính tổng, tổng bình phương độ dài) các đường đi xuất phát tại một đỉnh. Do cây là một đường thẳng nên việc đếm có thể thực hiện nhanh bằng thuật toán quy hoạch động.

Dộ phức tạp tính toán:  $O(N)$ .

- **Subtask 3:** 20% số test khác ứng với 20% số điểm thỏa mãn:  $w_i = 1$ .

Không cần quan tâm đến giới hạn truyền tải mà chỉ cần đếm (và tính tổng, tổng bình phương độ dài) các đường đi xuất phát tại một đỉnh. Dùng thuật toán quy hoạch động trên cây để giải quyết, việc chuyển nhãn sử dụng hằng đẳng thức  $(a + b)^2 = a^2 + b^2 + 2ab$ . Để tính toán cho từng đỉnh trên cây, có thể sử dụng kỹ thuật quay gốc của cây hoặc truyền thông tin xuống các nút con khi duyệt cây.

Dộ phức tạp tính toán:  $O(N)$ .

- **Subtask 4:** 16% số test khác ứng với 16% số điểm thỏa mãn:  $w_i \leq 1000$ ,  $\forall i = 1, 2, \dots, N - 1$ .

Xét lần lượt các giới hạn truyền tải theo thứ tự giảm dần, nạp các cạnh có cùng giới hạn truyền tải vào đồ thị. Đưa về đếm (và tính tổng, tổng bình phương độ dài) các đường đi xuất phát tại một đỉnh mà bắt buộc phải đi qua ít nhất một cạnh vừa nạp vào. Do các cạnh được nạp giảm dần theo giới hạn truyền tải, giới hạn truyền tải nhỏ nhất sẽ được biết. Việc đếm có thể bù trừ với bước đếm trước đó nên có thể dùng thuật toán quy hoạch động trên cây như subtask 3 để tính toán cho từng cây. Độ phức tạp tính toán:  $O(W \times N)$ .

- **Subtask 5:** 16% số test khác ứng với 16% số điểm thỏa mãn: Luôn có dây cáp nối giữa máy tính  $i$  và máy tính  $i + 1$ ,  $\forall i = 1, 2, \dots, N - 1$ .

*Cách 1:* Nạp dần từng cạnh vào theo thứ tự giảm dần giới hạn truyền tải. Khi nạp một cạnh vào, hai đoạn liên tiếp sẽ được nối với nhau thành một đoạn. Mỗi phần tử  $i$  trong đoạn đó sẽ được cập nhật kết quả, lượng tăng thêm này có thể tách thành ba phần là đơn vị, bội của  $i$  và bội của  $i^2$  nên có thể dùng ba cây quản lý đoạn (IT) để thực hiện nhanh.

Dộ phức tạp tính toán:  $O(N \times \log N)$ .

*Cách 2:* Sử dụng thuật toán chia để trị. Xét  $x$  là đỉnh chính giữa của các đường đi (các đường đi có độ dài lẻ thì xem như đỉnh chính giữa lệch về bên trái). Tải trọng nhỏ nhất của các đường đi xuất phát tại  $x$  đã được sắp xếp sẵn, bài toán đếm (và tính tổng, tổng bình phương độ dài) các đường đi xuất phát tại một đỉnh có thể giải quyết bằng thuật toán tìm kiếm nhị phân trên dãy đã sắp xếp.

Dộ phức tạp tính toán:  $O(N \times \log^2 N)$ .

- **Subtask 6:** 20% số test còn lại ứng với 20% số điểm:  $1 \leq N \leq 10^5$ .

Sử dụng thuật toán chia để trị tương tự như cách 2 của Subtask 5, tuy nhiên bước chia để trị phải thực hiện trên cây bằng thuật toán tìm tâm cây (centroid).

Dộ phức tạp tính toán:  $O(N \times \log^2 N)$ .

## Câu 6. Bài tập đếm giáng sinh (6,0 điểm)

- **Subtask 1:** Có 17,5% số test ứng với 17,5% số điểm thỏa mãn:  $M \leq 12$  và  $Q \leq 1024$ .

Với mỗi câu hỏi, ta duyệt vét cạn tất cả các phương án phân chia dãy có thể. Số phương án luôn không vượt quá  $2^M$ .

Dộ phức tạp:  $O(Q \times 2^M)$ .



- Subtask 2: 20% số test khác ứng với 20% số điểm thỏa mãn:  $X_i = Y_i = 1, \forall i = 1, 2, \dots, Q$ .

Với mỗi câu hỏi, ta cần tìm độ dài dãy phân chia có thứ tự từ điển lớn nhất. Ta sử dụng kỹ thuật tham lam kết hợp với bảng thừa.

Độ phức tạp:  $O((N + Q) \times \log N)$ .

- Subtask 3: 22,5% số test khác ứng với 22,5% số điểm thỏa mãn:  $L_i = 1, R_i = M$  và  $X_i, Y_i \leq 10^5, \forall i = 1, 2, \dots, Q$ .

Ta sử dụng kỹ thuật tương tự duyệt vét cạn để tìm ra  $10^5$  phương án chia có thứ tự từ điển lớn nhất và sử dụng cây tiền tố (Trie) để lưu lại, sau đó tìm tiền tố chung dài nhất bằng thuật toán tìm tiền tố chung gần nhất. Với mọi  $x \leq 10^5$  mã hóa của phương án có thứ tự từ điển thứ  $x$  và thứ  $x + 1$  sai khác không quá 26 phần tử, nên số nút của cây tiền tố không quá  $N + 26 \times 10^5$ .

Độ phức tạp:  $O((N + 26 \times 10^5 + Q) \times \log N)$ .

- Subtask 4: 22,5% số test khác ứng với 22,5% số điểm thỏa mãn:  $M \leq 2000$  và  $Q \leq 20000$ .

Ta sử dụng kỹ thuật thông thường để tìm dãy có thứ tự từ điển thứ  $X_i$  và  $Y_i$  rồi so sánh để tìm tiền tố chung dài nhất.

Độ phức tạp:  $O((N + Q) \times N)$ .

- Subtask 5: 17,5% số test còn lại ứng với 17,5% số điểm:  $1 \leq M, Q \leq 2 \times 10^5$ .

Ta chứng minh được rằng với mọi con liên tiếp độ dài lớn hơn hoặc bằng 90, số cách chia dãy luôn lớn hơn  $10^{17}$ . Do đó, ta dùng kỹ thuật bảng thừa để tìm ra phần đầu của dãy có thứ tự từ điển thứ  $X_i$ , sau đó tìm ra cách chia với 90 phần tử cuối bằng phương pháp quy hoạch động.

Độ phức tạp:  $O(Q \times (90 + \log n))$ .

### Lời giải chi tiết:

Gọi  $range(i)$  là chỉ số  $j$  lớn nhất để dãy  $C_i, C_{i+1}, \dots, C_j$  chứa các phần tử đôi một phân biệt.

Gọi  $F(\ell, r)$  là số cách chia dãy dãy  $C_\ell, C_{\ell+1}, \dots, C_r$  làm các dãy con liên tiếp chỉ chứa các phần tử đôi một phân biệt. Ta quy ước  $F(r+1, r) = 1$ , khi đó  $F(\ell, r) = \sum_{i=\ell}^{\min(r, range(\ell))} F(i+1, r)$ .

Vì  $C_i \neq C_{i+1}$  nên  $range(i) > i$  với mọi  $1 \leq i \leq N - 1$ . Do đó  $F(\ell, r) \geq F(\ell+1, r) + F(\ell+2, r)$  với mọi  $1 \leq \ell < r \leq N$ . Từ đây, ta dễ dàng chứng minh được  $F(\ell, r) \geq Fib_{r-\ell+1}$  với  $Fib_i$  là số thứ  $i$  trong dãy Fibonacci.

Do  $Fib_{90} > 10^{17}$ , ta thấy: mọi đoạn con liên tiếp có độ dài lớn hơn hoặc bằng 90 đều có trên  $10^{17}$  cách phân đoạn. Bởi vậy, việc xây dựng dãy mã hóa phân đoạn thứ  $x_i \leq 10^{17}$  sẽ diễn ra như sau: ta lần lượt xây dựng các đoạn của dãy theo thứ tự từ trái qua phải, khi nào phần chưa được chia đoạn còn nhiều hơn 90 phần tử, ta sẽ luôn chọn đoạn dài nhất có thể. Do đó, dãy mã hóa thứ  $x_i$  và dãy mã hóa thứ  $y_i$  sẽ chỉ khác nhau ở phần có 90 phần tử cuối cùng.

Tới đây, ta sử dụng kỹ thuật bảng thừa để tính nhanh số đoạn trong phần trước 90 phần tử cuối cùng (phần này chắc chắn thuộc tiền tố chung dài nhất của hai dãy), rồi tiếp tục xây dựng nốt cách chia đoạn cho 90 phần tử cuối cùng bằng phương pháp quy hoạch động thông thường.

----- HẾT -----

