

Tổng quan ngày thi thứ nhất

Bài	Tên file bài làm	Tên file dữ liệu	Tên file kết quả	Điểm
1	GAPTHU.*	GAPTHU.INP	GAPTHU.OUT	70
2	TETQUYMAO.*	TETQUYMAO.INP	TETQUYMAO.OUT	70
3	CHUTRINH.*	CHUTRINH.INP	CHUTRINH.OUT	60

Phần mở rộng * là PAS hay CPP tùy theo ngôn ngữ lập trình được sử dụng.

Giới hạn bộ nhớ là **1024MB** cho tất cả các bài.

Bài làm được chấm trên hệ thống VNOJ (<https://oj.vnoi.info>) với cấu hình bộ dịch như sau:

C++: -std=c++14 -pipe -O2 -s -static -lm -DTHEMIS -Wl,--stack,66060288

FPC: -dTHEMIS -O2 -XS -Sg -Cs66060288

LƯU Ý: Do các đội thi vào các thời điểm khác nhau, các bạn không được phát tán đề thi lên mạng hay chia sẻ cho bất kỳ ai cho đến hết ngày 9/2. Trong thời gian này, nghiêm cấm mọi hành vi trao đổi hay thảo luận về đề bài hoặc bài làm công khai hoặc với người bên ngoài.

Đề thi gồm 10 trang.

Hãy lập trình giải các bài toán sau đây.

Bài 1. GẤP THÚ BÔNG (70 điểm)

Chắc nhiều bạn đã biết, ngoài niềm đam mê bất tận với trà sữa, GSPVH còn rất ham thích gấp thú bông. Đã thành thói quen, mỗi khi ghé thăm trung tâm thương mại, khu vui chơi hay công viên nào mà có máy gấp thú, GS ắt phải trở tài với dăm ba đồng tiền lẻ. Nói là tiền lẻ thôi chứ đồng nào cũng màu hồng màu xanh cả. Trong cuộc sống hiện đại ngày nay, thú bông có nhiều lợi ích không thể nào bỏ qua. Những con gấu bông xinh xắn, dễ thương luôn khiến ai ai cũng cảm thấy nhẹ nhàng, êm ái mỗi khi ngắm nhìn chúng. Chúng giúp chúng ta xua tan bao áp lực bộn bề của cuộc sống vốn ngập tràn mỗi mết. Nhiều chú gấu bông còn có cặp má phính véo thì thích miễn chê rồi. Vào mùa đông lạnh giá, gấu bông như chiếc lò sưởi giúp chúng ta ấm áp cả thể chất lẫn tinh thần. Chỉ cần dang rộng vòng tay ôm lấy một chú gấu bông to lớn, những đêm dài lạnh lẽo nơi Hà Nội chỉ 8 độ C sẽ là những giấc ngủ ngon trong chiếc mền ấm êm. Kể ra thì gấu bông cũng có nhiều cái hơn một số loài gấu khác ấy chứ nhỉ. Thứ nhất, gấu bông không khiến bạn phải lo “đau ví”. Một khi đã gấp về rồi thì gấu sẽ luôn tươi tắn, vui cười và cặp má thì lúc nào cũng bầu bĩnh; trong khi với một số loài gấu khác, nếu không được chăm bón bằng trà sữa, bồng ngô CGV và bổ sung vitamin hên hò thường xuyên, gấu sẽ chuyển sang chế độ cau có, giận dữ. Và tất nhiên, chi phí cho trà sữa hay sweetbox này tốn hơn xèng gấp gấu bông từ máy rất nhiều. Như đã nói ở trên, trong đêm đông giá lạnh, gấu bông sẽ là nguồn nhiệt sưởi ấm tấm lòng, ấy nhưng, nếu ôm một số loài gấu khác, có khi bạn sẽ còn lạnh hơn

ấy chứ. Và quan trọng nhất, ngôn ngữ của loài gấu bông không có những từ “phụ tình”, “phối bạn” hay “thay lòng đổi dạ”. Gấu bông luôn sẵn sàng ở bên bạn mãi mãi, chứ ở một số loài gấu khác thì lắm lúc không phải như vậy.

Thôi nào, cuộc chia tay của những chú/con gấu không làm bằng bông thì hãy còn nhiều lắm. Giờ hãy quay về với cửa hàng gấp thú ưa thích của GSPVH đã.

Khu vui chơi tại chợ đêm Oileh ở quận Chải Hâu có một chiếc máy gấp thú với vô vàn con thú bông sắc sỡ. Để tri ân những tín đồ mê gấp thú và truyền cảm hứng gấp thú bông cho thế hệ Z, chợ đêm Oileh tung ra một chương trình ưu đãi hấp dẫn. Theo đó, giá mỗi lượt chơi gấp thú được mô tả bởi hai dãy số nguyên dương (s_1, s_2, \dots, s_n) và (p_0, p_1, \dots, p_n) . Giả sử σ là số tiền người chơi đã chi cho việc gấp thú trước đây, chi phí của lượt gấp thú kế tiếp được xác định như sau:

- ✿ Nếu $0 \leq \sigma < s_1$, giá tiền là p_0 .
- ✿ Nếu $s_1 \leq \sigma < s_2$, giá tiền là p_1 .
- ✿ Nếu $s_2 \leq \sigma < s_3$, giá tiền là p_2 .
- ✿ ...
- ✿ Nếu $s_{n-1} \leq \sigma < s_n$, giá tiền là p_{n-1} .
- ✿ Nếu $s_n \leq \sigma$, giá tiền là p_n .

Hai dãy số này thoả mãn các điều kiện $s_1 < s_2 < \dots < s_n$ và $p_0 > p_1 > p_2 > \dots > p_n$, ý tưởng là nếu bạn gấp thú càng nhiều, lượt gấp thú tiếp theo sẽ càng rẻ.

Ví dụ, giả sử $n = 3$ và $(s_1, s_2, s_3) = (20, 30, 40)$ và $(p_0, p_1, p_2, p_3) = (7, 5, 3, 2)$; giá mỗi lượt gấp thú như sau:

- ✿ Lượt gấp thú đầu tiên có giá $p_0 = 7$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 7 < s_1 = 20$.
- ✿ Lượt gấp thú thứ hai có giá $p_0 = 7$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 14 < s_1 = 20$.
- ✿ Lượt gấp thú thứ ba có giá $p_0 = 7$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 21 < s_2 = 30$.
- ✿ Lượt gấp thú thứ tư có giá $p_1 = 5$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 26 < s_2 = 30$.
- ✿ Lượt gấp thú thứ năm có giá $p_1 = 5$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 31 < s_3 = 40$.
- ✿ Lượt gấp thú thứ sáu có giá $p_2 = 3$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 34 < s_3 = 40$.
- ✿ Lượt gấp thú thứ bảy có giá $p_2 = 3$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 37 < s_3 = 40$.
- ✿ Lượt gấp thú thứ tám có giá $p_2 = 3$. Sau lượt này, số tiền người chơi đã chi cho việc gấp thú là $\sigma = 40 = s_3$.
- ✿ Kể từ lượt gấp thú thứ chín, giá mỗi lượt là $p_3 = 2$.

Trong máy gấp thú có một số con gấu bông. Mỗi con gấu bông có một giá trị riêng. Số lượng và giá trị của những con gấu bông bên trong máy được mô tả bởi m cặp số nguyên $(l_1, r_1), (l_2, r_2), \dots, (l_m, r_m)$ với ý nghĩa: Với mỗi bộ số (i, j) thỏa mãn $1 \leq i \leq m$ và $l_i \leq j \leq r_i$, trong máy có thêm một con gấu bông với giá trị j . Chú ý rằng, nếu có nhiều bộ số (i, j) với cùng một giá trị j thỏa mãn các điều kiện ở trên, trong máy sẽ có nhiều con gấu cùng giá trị. Như vậy, tổng số gấu bông trong máy là $(r_1 - l_1 + 1) + (r_2 - l_2 + 1) + \dots + (r_m - l_m + 1)$. Ví dụ, giả sử $m = 3$ và các bộ số lần lượt là $(2, 5), (8, 10), (10, 12)$. Khi đó, máy gồm 10 con gấu bông với giá trị lần lượt là 2, 3, 4, 5, 8, 9, 10, 10, 11, 12.

Với kinh nghiệm chinh chiến ở muôn vàn đấu trường gấp thú trên khắp mọi miền tổ quốc, GSPVH tự tin mình có thể gấp “bách phát bách trúng” (gấp phát nào trúng phát đó). Thế nhưng, để tránh việc nhân viên siêu thị phát hiện tài năng của GSPVH và đẩy GS ra xa đàn gấu thân yêu, GS luôn chọn gấp thú theo thứ tự giá trị từ nhỏ đến lớn. Nói cách khác, GS luôn chọn gấp con thú có giá trị nhỏ nhất trong số những con thú còn trong máy.

GSPVH đặt ra mục tiêu kiếm được số tiền t nhờ việc bán gấu gấp được (do nhà GS đã có đầy đủ gấu để ohm rùi). Do GS còn bận chuẩn bị đề thi PVHOI 3.0, GS muốn biết cần ít nhất bao nhiêu lượt gấp để số tiền kiếm được là ít nhất t (nói cách khác, tổng giá trị của những chú gấu gấp được trừ đi số tiền bỏ ra để chơi gấp thú không bé hơn t).

Để tăng độ khó cho bài toán, bạn được cho q số nguyên dương t_1, t_2, \dots, t_q . Với mỗi số t_i , hãy bạn cần cho GS biết số lần chơi gấp thú ít nhất để thu về được số tiền không ít hơn t_i .

DỮ LIỆU (vào từ file văn bản **GAPTHU.INP**):

- ✿ Dòng đầu tiên chứa ba số nguyên n, m và q ($1 \leq n, m, q \leq 10^5$).
- ✿ Dòng thứ hai chứa n số nguyên s_1, s_2, \dots, s_n ($0 < s_1 < s_2 < \dots < s_n < 4 \cdot 10^{18} + 1$).
- ✿ Dòng thứ ba chứa $n + 1$ số nguyên $p_0, p_1, p_2, \dots, p_n$ ($4 \cdot 10^{18} + 1 > p_0 > p_1 > p_2 > \dots > p_n > 0$).
- ✿ Trong m dòng tiếp theo, dòng thứ i chứa hai số nguyên l_i và r_i ($1 \leq l_i \leq r_i \leq 10^7$).
- ✿ Dòng cuối cùng chứa q số nguyên t_1, t_2, \dots, t_q ($1 \leq t_i \leq 4 \cdot 10^{18}$).

KẾT QUẢ (ghi ra file văn bản **GAPTHU.OUT**):

Gồm một dòng duy nhất chứa q số nguyên, số thứ i trong đó là số lượt chơi tối thiểu để GS kiếm được số tiền không ít hơn t_i . Nếu GS gấp hết toàn bộ thú trong máy mà vẫn không kiếm đủ số tiền như mục tiêu, in ra -1 .

SUBTASKS

- ✿ Subtask 1 (13 điểm): $m = n = 1, q \leq 1000$ và $r_1 \leq 1000$
- ✿ Subtask 2 (13 điểm): $m = n = 1$
- ✿ Subtask 3 (13 điểm): $n = 1$

- ✿ Subtask 4 (13 điểm): $n \leq 10$
- ✿ Subtask 5 (18 điểm): Không có ràng buộc gì thêm.

VÍ DỤ

GAPTHU . INP	GAPTHU . OUT
3 3 3	7 9 -1
20 30 40	
7 5 3 2	
2 5	
8 10	
10 12	
1 20 34	

GIẢI THÍCH

Cách tính giá mỗi lượt chơi cũng như giá trị của các con gấu bông trong máy gấp thú ở ví dụ trên đã được thể hiện trong phần đề bài.

- ✿ Sau 7 lần gấp thú, GS thu về số tiền là $2 + 3 + 4 + 5 + 8 + 9 + 10 = 41$ trong khi số tiền chi ra là 37, như vậy GS lời số tiền là $41 - 37 = 4$.
- ✿ Sau 9 lần gấp thú, GS thu về số tiền là $2 + 3 + 4 + 5 + 8 + 9 + 10 + 10 + 11 = 62$ trong khi số tiền chi ra là 42, như vậy GS lời số tiền là $62 - 42 = 20$.
- ✿ Nếu gấp hết toàn bộ 10 con gấu bông trong máy, GS cần chi số tiền là 44 đồng. Do tổng giá trị của các con gấu là $2 + 3 + 4 + 5 + 8 + 9 + 10 + 10 + 11 + 12 = 74$, GS chỉ kiếm được số tiền là $74 - 44 = 30$, thấp hơn chỉ tiêu là 34.

Bài 2. TRANG TRÍ NGÀY XUÂN (70 điểm)

Vậy là một mùa xuân Quý Mão lại về trên khắp muôn nơi. Xuân về là sự khởi đầu của bao chu kỳ mới: Đó là chu kỳ của một năm gồm 12 tháng, là chu kỳ 4 mùa xuân, hạ, thu, đông. Đó còn là chu kỳ sinh sôi, nảy nở của muôn loài. Xuân sang là lúc bao sức sống mới tràn về. Cỏ cây hoa lá chấm dứt thời kỳ ẩn mình cần cỗi trong cái lạnh tái tê, chuyển sang giai đoạn xanh tươi rực rỡ. Hoà chung vào sự chuyển đổi của đất trời, con người cũng trở nên vui tươi yêu đời hơn. Mỗi dịp “năm hết, Tết đến”, chúng ta tạm gác lại những lo toan bộn bề, những thử thách khó khăn còn sót lại của năm cũ, hoà mình vào hương xuân phơi phới, tích lũy năng lượng tràn trề và niềm tin về một năm mới ngập tràn tốt tươi. Bởi thế, cho dù cuộc sống ngày nay có gấp gáp và vội vã nhường nào, chúng ta vẫn ít nhiều chờ mong tới Tết và háo hức mỗi dịp xuân sang.

Ngày Tết đến trên khắp quê tôi, ngàn hoa thơm khoe sắc xinh tươi

Ngàn em thơ khoe áo mới, chạy tung tăng vui đón pháo xuân.

...

Ngày Tết đến ta chúc cho nhau, một năm thêm sung túc an vui

Người nông dân thêm lúa thóc, người thương gia mau phát tài.

(Ngày Tết quê em – Từ Huy)

Mùa xuân tươi đẹp, rực rỡ, rộn rã và đầy sức sống đến thế, ấy vậy mà có người lại chỉ muốn đông ở lại mãi mà chẳng thêm xuân sang. Đó chính là Nhật Quang. Nhật Quang yêu thích mùa đông là bởi vì, Nhật Quang yêu Tuyết, và chỉ có vào mùa đông thì Tuyết mới về. Không hiểu trái tim của Nhật Quang lạnh lẽo tới đâu mà anh chàng lại có gu cô nàng dưới 0 độ C như vậy. Với con người ta, xuân sang là lúc tâm hồn được sưởi ấm, được thoát khỏi cái lạnh tái tê để tung tăng bay nhảy. Nhưng với Nhật Quang, thấy Tuyết tan chảy rồi dần dần biến mất, chẳng khác nào xuân sang mang ngọn lửa đốt cháy rụi con tim của chàng. Dù chán ghét xuân sang, căm thù Tết đến, nhưng thôi để giống bao người ta, Quang đành phải dọn sạch Tuyết để đặt những khóm Trúc, cành Mai trang trí khu vườn.

Khu vườn nhà Nhật Quang có dạng lưới ô vuông gồm r hàng và c cột. Các hàng được đánh số từ 1 đến r và các cột được đánh số từ 1 đến c . Ô thuộc hàng i và cột j được ký hiệu là (i, j) . Nhật Quang dự định đặt hoa trang trí vào một số ô trong khu vườn. Vốn không muốn nhìn thấy Trúc hay Mai quá nhiều, Quang đặt ra luật sau cho việc đặt hoa trang trí: Nếu hai ô (x_1, y_1) và (x_2, y_2) thể hiện nước đi hợp lệ của quân mã trên bàn cờ, **cả hai ô này không thể cùng có hoa**. Nói cách khác, ít nhất một trong hai ô này phải không có hoa.

Ngoài ra, Nhật Quang còn có một dãy số đẹp a_1, a_2, \dots, a_r . Mỗi số trong dãy này có giá trị từ 0 đến 3. Để đặt hoa trang trí Như Ý Nhật Quang, số ô chứa hoa **trên mọi hàng** phải thỏa mãn hai điều kiện sau:

- ✿ Nếu $a_i < 3$, số ô có hoa trên hàng i phải đồng dư với a_i khi chia cho 3.
- ✿ Nếu $a_i = 3$, số ô có hoa trên hàng i có thể là một giá trị bất kỳ.

Ra Tết là lúc kỳ thi học sinh giỏi quốc gia đến thật gần, Nhật Quang muốn ôn lại các bài toán đếm. Và Quang đã nghĩ ra bài toán sau: Với mỗi ô (i, j) trong khu vườn, có bao nhiêu cách đặt hoa trang trí mà Nhật Quang cho là Như Ý, đồng thời ô (i, j) có được đặt hoa?

Nhắc lại, hai ô (x_1, y_1) và (x_2, y_2) thể hiện nước đi hợp lệ của quân mã trên bàn cờ khi và chỉ khi $(x_1 - x_2)^2 + (y_1 - y_2)^2 = 5$. Hai cách đặt hoa trang trí được coi là khác nhau khi và chỉ khi tồn tại một ô có hoa ở cách này, nhưng lại không có hoa ở cách khác.

DỮ LIỆU (vào từ file văn bản **TETQUYMAO.INP**):

- ✿ Dòng đầu tiên chứa hai số nguyên r và c ($1 \leq r \leq 60, 1 \leq c \leq 10$).
- ✿ Dòng thứ hai chứa r số nguyên a_1, a_2, \dots, a_r ($0 \leq a_i \leq 3$).

KẾT QUẢ (ghi ra file văn bản **TETQUYMAO.OUT**):

Gồm r dòng, mỗi dòng chứa c số nguyên. Số thứ j trên dòng thứ i là số cách đặt hoa trang trí Như Ý Nhật Quang mà ở đó ô (i, j) có hoa. Do kết quả có thể rất lớn, các bạn chỉ cần in phần dư của các số này khi chia cho $10^9 + 19972207$.

SUBTASKS

- ✧ Subtask 1 (14 điểm): $r \cdot c \leq 21$
- ✧ Subtask 2 (8.4 điểm): $c = 1$
- ✧ Subtask 3 (8.4 điểm): $c = 2$
- ✧ Subtask 4 (11.2 điểm): $c \leq 4$
- ✧ Subtask 5 (12.6 điểm): $c \leq 6$
- ✧ Subtask 6 (15.4 điểm): Không có ràng buộc gì thêm.

VÍ DỤ

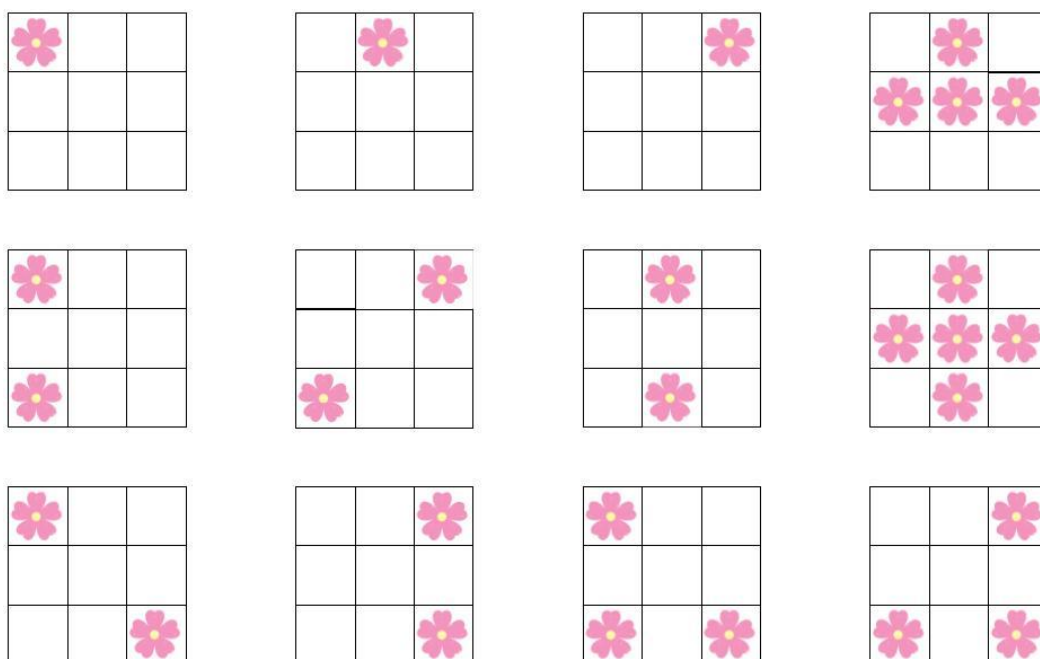
TETQUYMAO.INP	TETQUYMAO.OUT
3 3	4 4 4
1 0 3	2 2 2
	4 2 4

GIẢI THÍCH

Trong ví dụ trên, khu vườn nhà Nhật Quang có dạng lưới ô vuông gồm 3 hàng và 3 cột.

- ✧ Do $a_1 = 1$ nên số ô có hoa trên hàng 1 phải chia 3 dư 1 (tức phải bằng 1).
- ✧ Do $a_2 = 0$ nên số ô có hoa trên hàng 2 phải chia 3 dư 0 (tức bằng 0 hoặc 3).
- ✧ Do $a_3 = 3$ nên số ô có hoa trên hàng 3 có thể là một giá trị bất kỳ.

Hình vẽ dưới đây thể hiện tất cả 12 cách đặt hoa trang trí mà Nhật Quang cho là Như Ý.

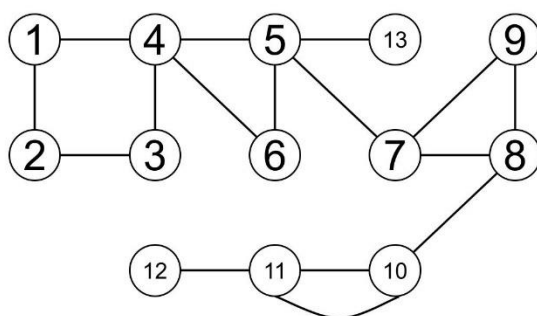


Bài 3. ĐẾM CHU TRÌNH (60 điểm)

Lý thuyết đồ thị là một mảng kiến thức vô cùng lớn và quan trọng bậc nhất của Toán rời rạc cũng như Lập trình thi đấu. Chính bởi vậy, đề thi học sinh giỏi quốc gia năm nào cũng chứa từ một tới hai bài toán đồ thị. Để chuẩn bị tốt nhất cho kỳ thi lớn sắp tới, hãy cùng nhau ôn lại đồ thị thông qua bài tập này nhé!

Trong bài tập này, ta chỉ xét các đồ thị vô hướng không chứa khuyên (cạnh nối một đỉnh với chính nó). Nói cách khác, đồ thị trong bài tập này có thể chứa cạnh lặp (tức giữa hai đỉnh có thể có nhiều cạnh nối), nhưng mỗi cạnh luôn nối hai đỉnh khác nhau.

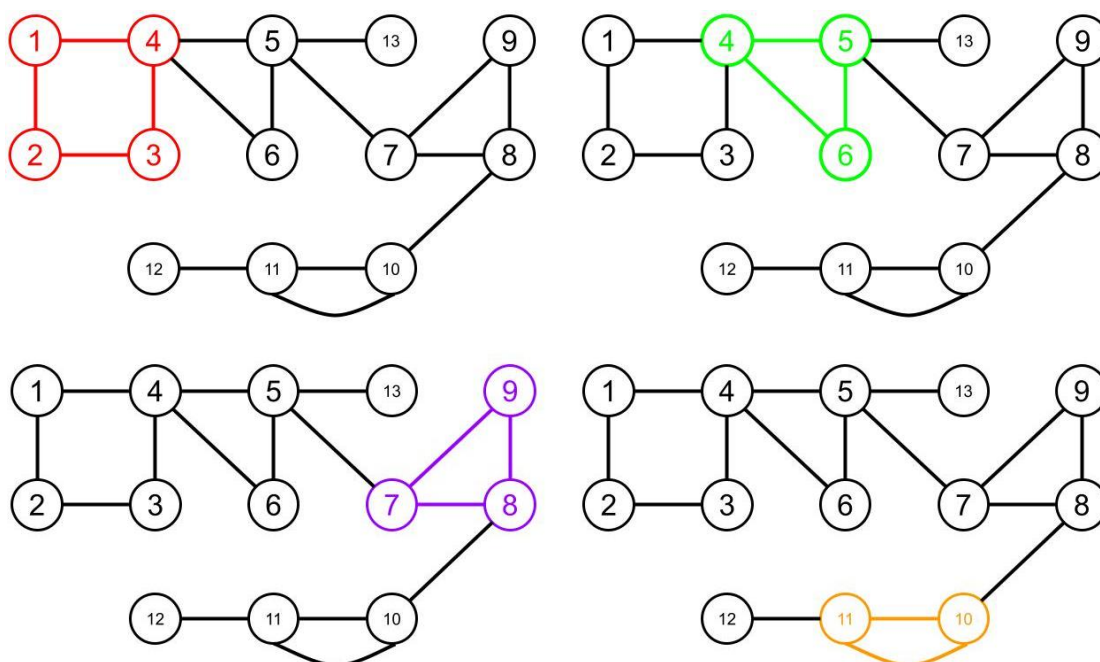
Dưới đây là hình ảnh của một đồ thị vô hướng gồm 13 đỉnh và 16 cạnh:



Giờ ta xét tới khái niệm *chu trình đơn*. Một *chu trình đơn* trong đồ thị vô hướng là một tập hợp con khác rỗng S của tập hợp các cạnh trên đồ thị thỏa mãn các tính chất dưới đây:

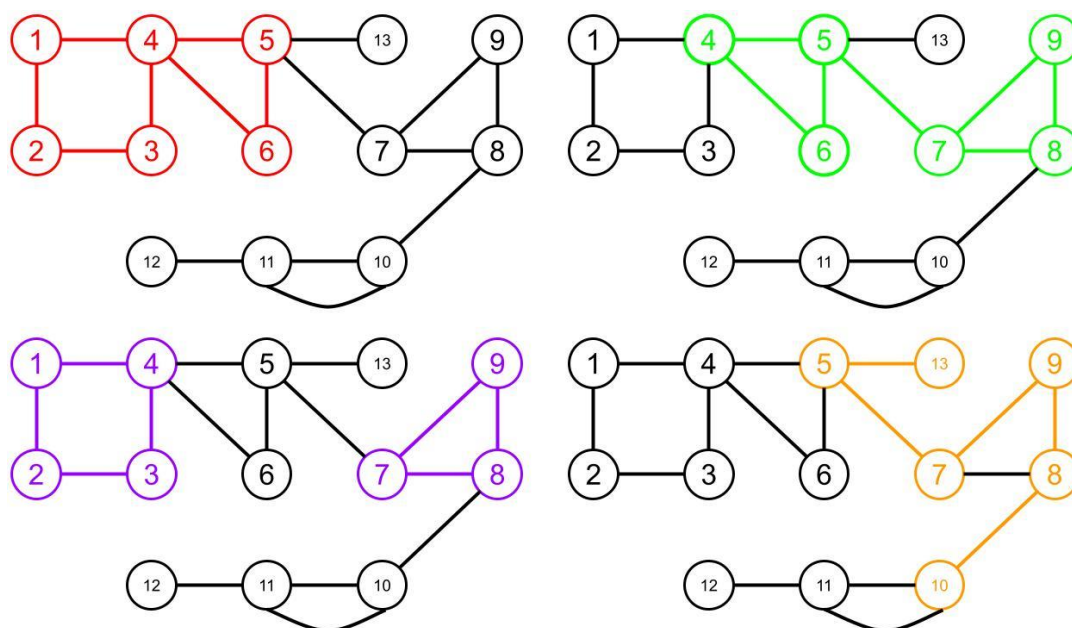
- ✿ Mọi đỉnh trên đồ thị kề với 0 hoặc 2 cạnh trong S .
- ✿ Các cạnh trong S liên thông với nhau. Nói cách khác, với mọi cặp đỉnh (u, v) của đồ thị sao cho cả u và v đều kề với ít nhất một cạnh thuộc S , luôn tồn tại đường đi giữa u và v chỉ đi qua các cạnh thuộc S .

Hình dưới đây thể hiện các chu trình đơn của đồ thị trên:



Hình dưới đây thể hiện một số tập hợp cạnh không thoả mãn các tính chất của chu trình đơn vì:

- Trong tập cạnh màu đỏ (hình góc trên bên trái), đỉnh 4 kề với 4 cạnh trong tập.
- Trong tập cạnh màu xanh (hình góc trên bên phải), đỉnh 5 kề với 3 cạnh trong tập.
- Tập cạnh màu tím (hình góc dưới bên trái) không thoả mãn điều kiện liên thông.
- Trong tập cạnh màu cam (hình góc dưới bên phải), đỉnh 10 kề với 1 cạnh trong tập.



Cây là một đồ thị vô hướng liên thông và không có chu trình. Một cây gồm n đỉnh luôn có chính xác $n - 1$ cạnh.

Trong bài tập này, bạn được cho một cây gồm n đỉnh. Các đỉnh được đánh số từ 1 đến n , các cạnh được đánh số từ 1 đến $n - 1$. Cạnh thứ i nối hai đỉnh x_i và y_i . Bạn cần trả lời q câu hỏi. Trong mỗi câu hỏi, bạn được cho sáu số nguyên $e_1, u_1, v_1, e_2, u_2, v_2$ thoả mãn $1 \leq e_1, e_2 < n, 1 \leq u_1, v_1, u_2, v_2 \leq n, e_1 \neq e_2, u_1 \neq v_1$ và $u_2 \neq v_2$. Bạn cần xây dựng một đồ thị mới bằng cách lấy cây ban đầu, xoá đi hai cạnh e_1 và e_2 đồng thời thêm hai cạnh (u_1, v_1) và (u_2, v_2) , sau đó đếm số chu trình đơn trong đồ thị mới này. Chú ý rằng, cây ban đầu không bị biến đổi trong suốt các câu hỏi. Đồ thị trong mỗi câu hỏi đều được xây dựng từ cây ban đầu, với hai cạnh bị xoá đi và hai cạnh được thêm mới.

Để giảm kích thước file dữ liệu và file kết quả, thay vì cho trực tiếp các tham số và yêu cầu in kết quả của mọi câu hỏi, bạn cần xét đoạn mã nguồn sau:

```
const int MOD = (int)1e9 + 19972207;
int cur = seed;
int getRandom(int n) {
    cur = (1LL * mul * cur + add) % MOD;
    return 1 + cur % n;
}
```



```

int gspvhcute(void) {
    int res = 0;
    for (int i = 1; i <= q; i++) {
        int e1 = getRandom(n - 1);
        int u1 = getRandom(n); int v1 = getRandom(n);
        int e2 = getRandom(n - 1);
        int u2 = getRandom(n); int v2 = getRandom(n);
        int tmp = e1 == e2 || u1 == v1 || u2 == v2
            ? MOD - 1 : query(e1, u1, v1, e2, u2, v2);
        res = (227LL * res + tmp) % MOD;
    }
    return res;
}

```

Trong đoạn mã nguồn trên, hàm `query(e1, u1, v1, e2, u2, v2)` nhận vào sáu tham số và trả về một số nguyên, là số chu trình đơn trong câu hỏi với các tham số $e_1, u_1, v_1, e_2, u_2, v_2$ theo định nghĩa ở trên. Bạn được cho biết giá trị của các biến `seed`, `mul`, `add` trong đoạn mã nguồn ở trên; hãy xác định kết quả trả về của hàm `gspvhcute`.

Vui lòng xem kỹ phần giải thích để hiểu rõ hơn về đoạn mã nguồn này.

DỮ LIỆU (vào từ file văn bản **CHUTRINH.INP**):

- ✳ Dòng đầu tiên chứa số nguyên θ ($1 \leq \theta \leq 6$) là số thứ tự của subtask chứa test này.
- ✳ Dòng thứ hai chứa số nguyên n ($2 \leq n \leq 150000$).
- ✳ Trong $n - 1$ dòng tiếp theo, dòng thứ i chứa hai số nguyên x_i và y_i ($1 \leq x_i, y_i \leq n$) mô tả cạnh thứ i của cây ban đầu. Dữ liệu vào đảm bảo $n - 1$ cạnh này tạo thành cây.
- ✳ Dòng cuối cùng chứa bốn số nguyên $q, seed, mul, add$ ($1 \leq q \leq 5555555$, $1 \leq seed, mul, add \leq 10^9$) lần lượt là số câu hỏi và giá trị của các biến trong đoạn mã nguồn ở trên.

KẾT QUẢ (ghi ra file văn bản **CHUTRINH.OUT**):

In ra một số nguyên duy nhất là kết quả trả về của hàm `gspvhcute`.

SUBTASKS

- ✳ Subtask 1 (12 điểm): $n \leq 16$ và $q \leq 89$
- ✳ Subtask 2 (12 điểm): Trong cây ban đầu, mỗi đỉnh kề với ít hơn ba đỉnh khác.
- ✳ Subtask 3 (7.2 điểm): $n \leq 50$ và $q \leq 7000$
- ✳ Subtask 4 (7.2 điểm): $n \leq 3000$ và $q \leq 40000$
- ✳ Subtask 5 (9.6 điểm): $q \leq 200000$
- ✳ Subtask 6 (12 điểm): Không có ràng buộc gì thêm.

VÍ DỤ

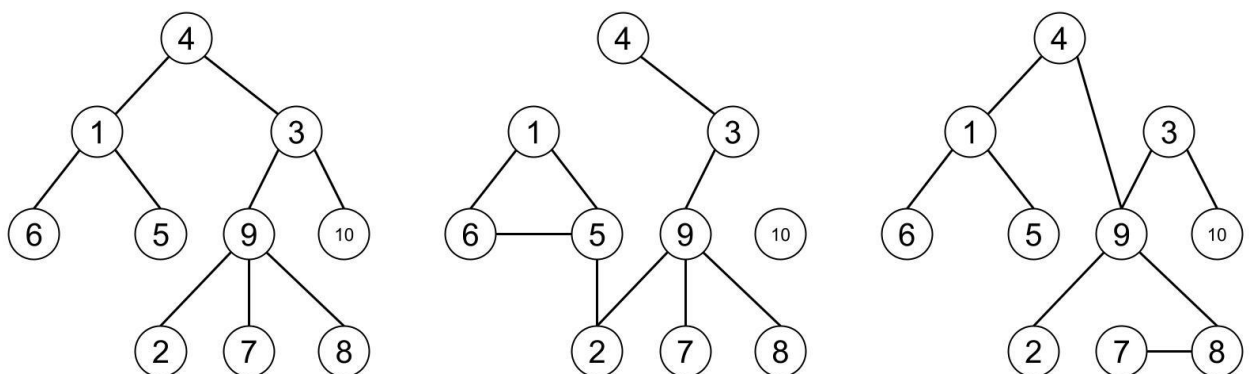
CHUTRINH.INP	CHUTRINH.OUT
1	51528
10	
1 4	
8 9	
7 9	
3 4	
3 9	
2 9	
3 10	
1 5	
1 6	
3 22 7 1997	

GIẢI THÍCH

Trong ví dụ trên, $q = 3$. Vòng lặp **for** trong đoạn mã nguồn trên diễn ra như sau:

- ✳ Tại $i = 1$, ta có $(e_1, u_1, v_1, e_2, u_2, v_2) = (1, 5, 6, 7, 2, 5)$. Hàm **query** được gọi và trả về 1. Do đó giá trị biến **res** trở thành $(227 \cdot 0 + 1) \% MOD = 1$.
- ✳ Tại $i = 2$, ta có $(e_1, u_1, v_1, e_2, u_2, v_2) = (4, 9, 4, 3, 7, 8)$. Hàm **query** được gọi và trả về 0. Do đó giá trị biến **res** trở thành $(227 \cdot 1 + 0) \% MOD = 227$.
- ✳ Tại $i = 3$, ta có $(e_1, u_1, v_1, e_2, u_2, v_2) = (5, 4, 4, 4, 7, 3)$. Do $u_1 = v_1$ nên giá trị biến **tmp** là $MOD - 1$. Giá trị biến **res** lúc này là $(227 \cdot 227 + MOD - 1) \% MOD = 51528$.

Trong hình vẽ dưới đây, từ trái qua phải lần lượt là cây lúc ban đầu, đồ thị được xét bởi hàm **query** khi $i = 1$ và khi $i = 2$.



❧ HẾT ❧