

MỤC LỤC

BÀI 1. GIẢI CỨU CON TIN	4
BÀI 2. CEZAR	4
BÀI 3. BFS	5
BÀI 4. BFS2	6
BÀI 5. JOURNEY	7
BÀI 6. RINGS	7
BÀI 7. PATROL	8
BÀI 8. MÊ CUNG	9
BÀI 9. ĐẦY HỘP	10
BÀI 10. TÔ MÀU	11
BÀI 11. WORMS	12
BÀI 12. XE ĐỊA HÌNH	13
BÀI 13. DFS	13
BÀI 14. CÁC BĂNG ĐĂNG Ở CHICAGO	14
BÀI 15. TALLER	14
BÀI 16. ĐI BỘ NGẮM CẢNH	15
BÀI 17. THREE MUSKETEERS	16
BÀI 18. NHỮNG CHÚ LÍNH CHÌ DỪNG CẢM	17
BÀI 19. NHỮNG CHÚ LÍNH CHÌ 2015	17
BÀI 20. BFS1	18
BÀI 21. ĐƯỜNG ĐI DÀI NHẤT CHIA HẾT CHO K	19
BÀI 22. ĐẾM HÌNH	19
BÀI 23. TRUYỀN TIN	20
BÀI 24. BIẾN ĐỔI SỐ	21
BÀI 25. LÂU ĐÀI	22
BÀI 26. MÊ CUNG	23
BÀI 27. THOÁT KHỎI MÊ CUNG	24
BÀI 28. KNIGHT07	25
BÀI 29. KNIGHT08	26
BÀI 30. ROADS	26

BÀI 31. ROADS2.....	27
BÀI 32. SORTING	28
BÀI 32. SỨC MẠNH PHI THƯỜNG	28
BÀI 33. CHỊ HẰNG VÀ CHÚ CUỘI.....	29
BÀI 34. KNIGHT09	30
BÀI 35. BIẾN ĐỔI BẢNG	31
BÀI 36. ĐỒ THỊ CÓ CẠNH BIẾN MẤT.....	31
BÀI 37. TRÒ CHƠI.....	32
BÀI 38. PHỞ NAM ĐỊNH	33
BÀI 38. CHÈO THUYỀN	34
BÀI 39. TRỜI CHO.....	34
BÀI 40. CPATH	36
BÀI 41. ANT	37
BÀI 42. DIJ.....	38
BÀI 43. ĐỒNG BẠC CỔ	38
BÀI 44. PENCIL	39
BÀI 45. INCPATH.....	40
BÀI 46. BIN LADEN	41
BÀI 46. SWITCH THE LAMP ON	42
BÀI 47. THANG MÁY	43
BÀI 48. Đảo rỗng	44
BÀI 49. ROADBLOCKS	45
BÀI 50. BUGS.....	46
BÀI 51. Khu vui chơi	47
BÀI 52. TOURS13	48
BÀI 53. PARTY2	49
BÀI 54. THỨ 6 NGÀY 13	49
BÀI 55. DU LỊCH	50
BÀI 56. Thăm bạn	52
BÀI 57. CẢNH SÁT	53
BÀI 57. WEATHER.....	54

BÀI 58. LIÊN THÔNG	55
BÀI 59. MA CÀ RỒNG	56
BÀI 60. HD nổi loạn.....	57
BÀI 61. JOB	57
BÀI 62. HOLES	58
BÀI 62. Shopping	59
BÀI 63. PATH1	60
BÀI 64. Vé xe miễn phí.....	61
BÀI 65. Vé xe miễn phí.....	62
BÀI 66. Mario cứu công chúa	63
BÀI 67. Cấm đường	64
BÀI 68. ROBOT	65
BÀI 69. Xây dựng thư viện	67
BÀI 70. PATH.....	68
BÀI 71. SPECIAL WALK	69
BÀI 72. Unique Trip.....	69
BÀI 73. MAGIC LABYRINTH.....	70
BÀI 74. GCDGRAPH	70
BÀI 75. The transportation network.....	71
BÀI 76. 999 đóa hồng.....	72
BÀI 77. Journey Scheduling.....	73
BÀI 78. Chocolate Journey	74
BÀI 79. Buses Journey	75
BÀI 80. Grid	76
BÀI 81. Đi ngược chiều.....	76

BÀI 1. GIẢI CỨU CON TIN

Trước sức ép của báo chí và dư luận, cơ quan mật vụ Anh MI6 đã phải sử dụng toàn bộ lực lượng của mình hòng giải cứu con tin đang bị bắt cóc tại Iraq. Đội đặc nhiệm cần tổ chức một đội càng nhiều người càng tốt, đột nhập vào căn cứ, tiêu diệt bọn khủng bố. Để cuộc giải cứu diễn ra tốt đẹp cần phải có sự phối hợp ăn ý giữa các điệp viên. Rất tiếc rằng không phải điệp viên nào cũng từng biết nhau (do tính bí mật của công việc) và đồng thời nếu A quen biết B thì B cũng quen biết A. Người ta yêu cầu mỗi thành viên trong đội giải cứu phải quen ít nhất K thành viên khác trong đội. Hãy lựa chọn đội gồm nhiều thành viên nhất để hoàn thành nhiệm vụ

Dữ liệu: Vào từ file văn bản **RESCUE.INP**

- Dòng đầu tiên ghi hai số N, M, K là số điệp viên, số mối quan hệ quen biết và số người trong đội mà một thành viên phải quen biết. ($N < 10001, K < 15001$)
- Tiếp theo là m dòng, mỗi dòng ghi hai cặp số (u,v) cho biết hai điệp viên u và v là quen nhau.

Kết quả: Ghi ra file **RESCUE.OUT** một số nguyên là số lớn nhất có thể của số thành viên trong đội giải cứu con tin.

Ví dụ:

RESCUE.INP	RESCUE.OUT
4 4 2 1 2 1 4 2 3 2 4	3

BÀI 2. CEZAR

Tại HT Land, có tất cả n thượng nghị sĩ ở trong n ngôi biệt thự (đánh số từ 1 đến n), giữa 2 ngôi nhà có một đường đi duy nhất: đường đi trực tiếp hoặc không trực tiếp (đi qua các con đường khác). Tất cả các ngôi nhà đều đi được đến nhau.

Mỗi thượng nghị sĩ khi đi từ nhà mình đến thượng viện, phải trả 1 USD khi đi qua một con phố (phố = đường nối trực tiếp 2 nhà bất kỳ). HT – người đứng đầu thượng viện - đã nghĩ cách làm sao cho số tiền mà các thượng nghị sĩ phải trả là tối thiểu. Vì vậy, HT quyết định

- Có k con phố miễn phí (thượng nghị sĩ sẽ không phải trả tiền khi đi trên con phố này)
- Đặt tòa nhà thượng viện ở một trong n ngôi nhà

Bạn hãy viết chương trình tính xem chi phí tối thiểu là bao nhiêu?

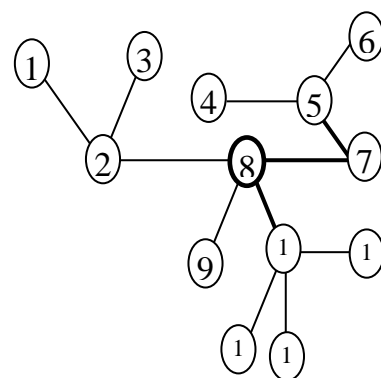
Dữ liệu: Vào từ file văn bản **CEZAR.INP**

- Dòng 1: Số nguyên n và k tương ứng là số ngôi nhà ở HT land và số đường phố miễn phí. ($1 < n \leq 10\,000, 0 < k < n$)
- $n-1$ dòng tiếp theo, mỗi dòng chứa 2 số u, v ($1 \leq u \neq v \leq n$) có nghĩa là có con phố nối ngôi nhà u và ngôi nhà v

Kết quả: Ghi ra file **CEZAR.OUT** một số nguyên duy nhất là chi phí tối thiểu phải trả

Ví dụ

CEZAR.INP	CEZAR.OUT	Giải thích
13 3 1 2 2 3 2 8 7 8 7 5 5 4 5 6 8 9 8 10 10 11 10 12 10 13	11	Có nhiều phương án giải quyết: Đây là 1 phương án: <ul style="list-style-type: none"> - 5-7, 7-8, 8-10 là những đường miễn phí - 8 là thượng viện Chi phí tối thiểu là: 11 .



BÀI 3. BFS

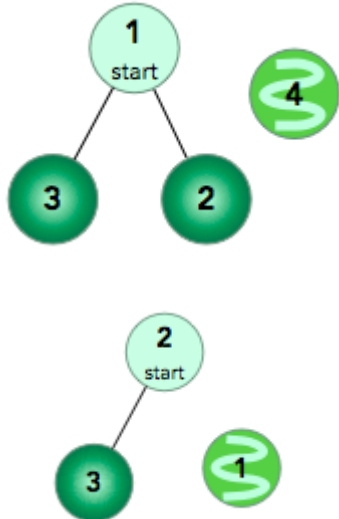
Cho đồ thị vô hướng n đỉnh đánh số từ 1 đến n có m cạnh có độ dài bằng nhau và bằng 6. Hãy cho biết độ dài đường đi ngắn nhất từ đỉnh s đến tất cả các đỉnh còn lại, nếu từ s không thể đến được đỉnh nào đó thì in -1.

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

- Dòng 1: q ($1 \leq q \leq 10$) là số test
- Mỗi test bắt đầu bằng:
 - o Dòng đầu tiên n, m ($2 \leq n \leq 1000, 1 \leq m \leq \frac{n*(n-1)}{2}$)
 - o m dòng tiếp theo, mỗi dòng 2 số nguyên dương u, v mô tả cạnh nối hai đỉnh u và v
 - o Dòng cuối cùng là số nguyên dương s , là đỉnh xuất phát.

Kết quả: ghi ra file văn bản **BFS.OUT** trên q dòng, mỗi dòng ghi $n-1$ số, tương ứng là độ dài đường đi ngắn nhất từ đỉnh s đến các đỉnh $1, 2, 3, \dots, n$ (không tính đỉnh s). In -1 nếu không tồn tại đường đi

Ví dụ:

BFS.INP	BFS.OUT	
2 4 2 1 2 1 3 1 3 1 2 3 2	6 6 -1 -1 6	

BÀI 4. BFS2

Cho đồ thị có n đỉnh đánh số từ 1 đến n .

có m truy vấn dạng như sau:

- 1 x : cho biết độ dài đường đi ngắn nhất từ 1 đến x .
- 2 x y : Thêm cung $x \rightarrow y$

Dữ liệu: vào từ file văn bản **BFS2.INP**

- Dòng đầu tiên n, m ($1 \leq n \leq 1000, 1 \leq m \leq 300\,000$)
- m dòng tiếp theo mô tả m truy vấn dạng:
 - o 1 x ($1 \leq x \leq n$)
 - o 2 x y ($1 \leq x, y \leq n$)

Kết quả: ghi ra file văn bản **BFS2.OUT** trên một số dòng tương ứng với truy vấn 1. Nếu không có đường đi từ 1 đến x thì ghi -1

Ví dụ:

BFS2.INP	BFS2.OUT
4 7 1 4 2 1 2 2 2 3 2 3 4 1 4 2 2 4 1 4	-1 3 2

BÀI 5. JOURNEY

Có N phi hành gia đánh số từ 0 đến $n - 1$ và chính phủ muốn chọn ra 2 phi hành gia không cùng 1 thành phố. Chính phủ không biết rõ mỗi phi hành gia ở thành phố nào mà chỉ biết p mối quan hệ u, v cho biết phi hành gia u và v ở cùng một thành phố.

Yêu cầu: Hãy đếm số cách chọn thỏa mãn.

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

- Dòng 1: N, P ($1 \leq N \leq 10^5, 1 \leq P \leq 10^4$)
- P dòng tiếp theo, mỗi dòng hai số nguyên u, v

Kết quả: ghi ra file văn bản **JOURNEY.OUT** một số duy nhất là đáp án?

Ví dụ:

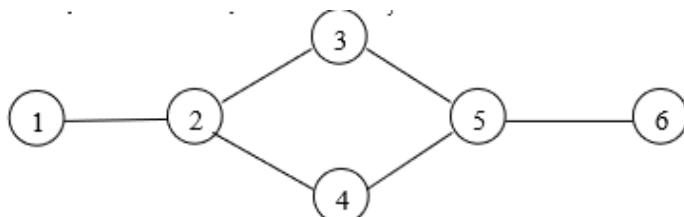
JOURNEY.INP	JOURNEY.OUT
5 3 0 1 2 3 0 4	6

JOURNEY.INP	JOURNEY.OUT
4 1 0 2	5

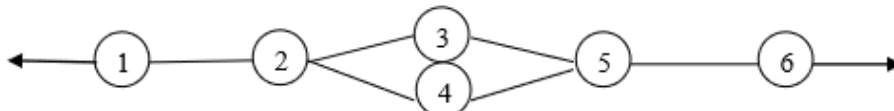
BÀI 6. RINGS

Cho đồ thị vô hướng N đỉnh, M cạnh có độ dài bằng nhau. Tom và Jerry chọn hai đỉnh A và B (Tom chọn đỉnh A , Jerry chọn đỉnh B), cả hai sẽ kéo 2 đỉnh theo các hướng ngược nhau cho đến khi còn có thể. Kết quả, một số đỉnh (hoặc tất cả) sẽ thẳng hàng với A và B , và số còn lại sẽ bị treo (không thẳng hàng).

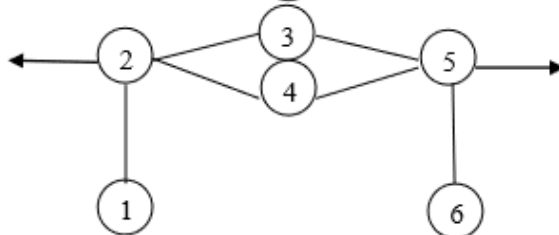
Ví dụ: Cho đồ thị như hình 1, có $N = 6$ và $M = 6$.



(Hình 1)



(Hình 2)



(Hình 3)

- Nếu Tom và Jerry chọn hai đỉnh $A = 1$ và $B = 2$ và kéo, khi đó tất cả các đỉnh đều thẳng hàng, không có đỉnh treo. (Hình 2)
- Nếu Tom và Jerry chọn hai đỉnh $A = 2$ và $B = 5$ và kéo, khi đó 3, 4 sẽ thẳng hàng với 2 và 5. 1 và 6 là những đỉnh treo. (Hình 3).

Yêu cầu: Viết chương trình xác định số đỉnh treo, khi Tom và Jerry kéo 2 đỉnh A và B.

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

- Dòng 1: 2 số nguyên N và M ($5 < N \leq 5000$)
- m dòng sau, mỗi dòng chứa 2 số x, y , nghĩa là có cạnh nối giữa hai đỉnh x và y .
- 2 dòng cuối, mỗi dòng chứa 2 số A và B, tương ứng là 2 test của bài toán.

Kết quả: ghi ra file văn bản **RINGS.OUT** trên cùng một dòng, 2 số nguyên theo thứ tự là đáp số của 2 test trong file input. 2 số này ghi cách nhau một dấu cách.

Ví dụ:

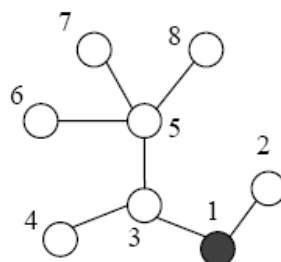
RINGS.INP	RINGS.OUT
6 6 1 2 2 3 2 4 3 5 4 5 5 6 1 6 2 5	0 2

BÀI 7. PATROL

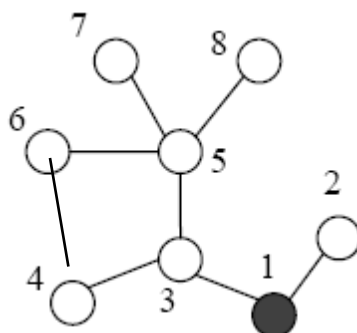
Một thành phố có N ngôi làng được đánh số từ 1 đến N . Có $N - 1$ con đường nối giữa các ngôi làng. Mỗi con đường nối 2 ngôi làng, và luôn có đường đi từ 2 ngôi làng bất kỳ bằng cách sử dụng các con đường đó. Độ dài của mỗi ngôi làng là 1 đơn vị thời gian.

Để đảm bảo an toàn cho mọi người trong thành phố, hàng ngày cảnh sát thành phố phải đi tuần tra dọc trên các con đường. Đồn cảnh sát được đặt tại làng 1, vì vậy hành trình của cảnh sát phải bắt đầu từ làng 1 và quay trở về làng 1 khi kết thúc một ngày.

Ví dụ dưới đây cho ta kết quả đi tuần mỗi ngày của cảnh sát là mất 14 đơn vị thời gian.



Để tiết kiệm thời gian đi tuần của cảnh sát, thành phố quyết định xây dựng thêm 1 đường tắt kết nối giữa 2 ngôi làng. Sau khi xây xong đường tắt, đội tuần tra chỉ được đi lại con đường tắt đúng 1 lần.



Hình trên là ví dụ về việc xây thêm đường tắt. Trong ví dụ, một đường tắt được xây dựng nối ngôi làng thứ 4 với ngôi làng thứ 6 và tổng đơn vị thời gian đi tuần tra 1 ngày là 12.

Viết chương trình tìm cách xây dựng thêm 1 đường tắt sao cho tổng đơn vị thời gian đi tuần trong 1 ngày của cảnh sát là ít nhất có thể.

Dữ liệu: Vào từ file văn bản **PATROL.INP**:

- Dòng đầu tiên chứa số nguyên N ($2 \leq N \leq 10^5$).
- $N - 1$ dòng tiếp theo, mỗi dòng chứa hai số nguyên A, B ($1 \leq A, B \leq N$), tức là có đường nối trực tiếp từ ngôi làng A tới ngôi làng B .

Kết quả: Ghi ra file văn bản **PATROL.OUT** một số nguyên dương duy nhất là tổng chi phí thời gian ít nhất sau khi xây dựng 1 đường tắt.

Ví dụ:

PATROL.INP	PATROL.OUT
8	11
1 2	
3 1	
3 4	
5 3	
7 5	
8 5	
5 6	

BÀI 8. MÊ CUNG

Mê cung có dạng một hình chữ nhật kích thước $m \times n$ được chia ra thành lưới ô vuông cạnh độ dài 1 bằng các đường song song với các cạnh. Mỗi ô vuông của lưới hoặc là ô cấm hoặc là ô tự do. Từ một ô tự do có thể di chuyển sang các ô tự do có chung cạnh với nó. Không được phép di chuyển vượt khỏi biên của mê cung. Mê cung được thiết kế khá đặc biệt: Giữa hai ô tự do bất kỳ chỉ duy nhất có một cách di chuyển từ ô này

đến ô kia. Tại tâm của mỗi ô tự do đều có một cái móc. Trong mê cung có hai ô tự do đặc biệt mà nếu bạn nối được hai cái móc ở hai ô đó bởi một sợi dây thừng thì cánh cửa bí mật của mê cung sẽ tự mở. Vấn đề đặt ra là phải chuẩn bị sợi dây thừng với độ dài ngắn nhất đảm bảo cho dù hai ô đặc biệt có ở vị trí nào trong mê cung bạn vẫn có thể nối được các cái móc ở hai ô đó bằng sợi dây thừng đã chuẩn bị.

Dữ liệu: Vào từ file văn bản **LABYR.INP**:

- Dòng đầu tiên chứa hai số m, n ($3 \leq m, n \leq 1000$)
- Các dòng tiếp theo mô tả mê cung. Dòng thứ i trong số m dòng tiếp theo chứa n ký tự, mỗi ký tự chỉ là '#' hoặc '.', trong đó ký tự '#' cho biết ô ở vị trí tương ứng là ô cấm, còn ký tự '.' cho biết ô ở vị trí tương ứng là ô tự do.

Kết quả: Ghi ra file văn bản **LABYR.OUT** độ dài (tính bởi số ô) của sợi dây thừng cần chuẩn bị

Ví dụ:

LABYR.INP	LABYR.OUT
3 3 ### #.# ###	0

LABYR.INP	LABYR.OUT
7 6 ##### #.#.### #.#.### #.#.### #.#.### #.....# #####	8

BÀI 9. ĐẨY HỘP

Cho đồ thị liên thông gồm N nút được đánh số từ 1 đến N và M cung có hướng. Trên N nút này, tại mỗi nút có không quá một khối hộp, có nút không có khối hộp nào. Một bước đẩy được coi là đúng nếu ta đẩy một khối hộp từ vị trí hiện tại đi theo cung có hướng tương ứng nào đó đến một nút rỗng. Một nút rỗng là một nút hoặc là không chứa gì hoặc nút đó là nút 1. Hãy tìm lịch trình đẩy tất cả các hộp về nút 1 sao cho số bước thực hiện là nhỏ nhất.

Dữ liệu: Vào từ file văn bản **PUSHBOX.INP**

- Dòng đầu tiên là ba số nguyên dương N, M và K trong đó K là số hộp. ($1 \leq N \leq 10\,000, 0 \leq M \leq 50\,000, 0 \leq K \leq N$)
- Dòng thứ hai ghi K số khác nhau là số hiệu của K ô có hộp.
- M dòng tiếp, mỗi dòng ghi hai số u, v với ý nghĩa có cung nối từ u đến v .

Kết quả: Ghi ra file văn bản **PUSHBOX.OUT**

- Dòng đầu tiên ghi ra số S là số bước đẩy ít nhất ta cần phải thực hiện.
- S dòng tiếp theo mô tả quá trình đẩy hộp. Trong đó mỗi dòng ghi hai số x, y với ý nghĩa cần phải đẩy hộp từ x đến y .

Ví dụ:

PUSHBOX.INP	PUSHBOX.OUT
4 3 3	4
2 3 4	2 1
3 1	4 2
2 1	2 1
4 2	3

BÀI 10. TÔ MÀU

Cho N số từ 1 đến N được viết vào các ô của bảng kích thước $2*N$ sao cho mỗi số xuất hiện đúng 2 lần trong bảng và mỗi cột chứa 2 số khác nhau: Ví dụ:

1	5	3	1	5
4	2	2	4	3

Chúng ta sẽ tô màu mỗi ô bằng màu trắng hoặc đen sao cho 2 ô ở cùng một cột sẽ tô màu khác nhau và các ô chứa các số bằng nhau cũng sẽ tô màu khác nhau: Ví dụ trên chúng ta có thể tô màu nhau sau:

1	5	3	1	5
4	2	2	4	3

Yêu cầu: Có bao nhiêu cách tô màu như vậy?

Dữ liệu: Vào từ file văn bản **COLOUR.INP**

- Dòng 1: Ghi số nguyên dương N ($2 \leq N \leq 100$)
- 2 dòng tiếp theo, mỗi dòng ghi N số nguyên trong đoạn $[1, N]$, các số này mô tả các số trên bảng $2*N$

Kết quả: Ghi ra file văn bản **COLOUR.OUT** một số duy nhất là số cách sử dụng hai màu để tô bảng $2*N$.

Ví dụ:

COLOUR.INP	COLOUR.OUT
5 1 5 3 1 5 4 2 2 4 3	4

BÀI 11. WORMS

Ở vùng đất Tpland có nhiều ngôi nhà của giun sinh sống. Không phải tất cả các ngôi nhà đều có giun ở, và mỗi ngôi nhà cũng chỉ có nhiều nhất một con giun sinh sống. Một vài ngôi nhà có đường nối với nhau. Đối với bất kỳ cặp của hai nhà khác nhau tồn tại đúng một tuyến đường bao gồm đường nối, do đó không có đường đi lặp đi lặp lại trên tuyến đường này.// đề thi liên thông không chu trình

Một ngày đẹp trời, tất cả các con giun ở Tpland quyết định gặp nhau ở một trong các ngôi nhà. Giun đi bộ dọc theo các con đường trong đất mất đúng một giờ (từ ngôi nhà đầu đường đến ngôi nhà cuối đường). Giun có ý định tiếp tục đi cho đến thời điểm tất cả chúng ở trong cùng một nhà (Giun cần phải ở cùng một nhà, chính xác tại cùng một thời điểm).

Thật không may, Giun không thể dự đoán có thể mất một thời gian bao nhiêu để gặp nhau. Đôi khi thậm chí không thể đáp ứng. Đó là lý do tại sao Giun hỏi bạn để được trợ giúp, xác minh xem cuộc gặp mặt của họ có thể được sắp xếp và nếu có, trong trường hợp tốt nhất, sẽ mất bao nhiêu thời gian?

Dữ liệu: Vào từ file văn bản **WORMS.INP**

- Dòng 1: Gồm 2 số nguyên dương n và m ($2 \leq n \leq 50.000$) ($1 \leq m \leq 50.000$) tương ứng là số ngôi nhà (đánh số từ 1 đến n) và số con đường ở Tpland.
- M dòng tiếp theo, mỗi dòng ghi 2 số nguyên a và b ($1 \leq a, b \leq n$) là có đường đi giữa 2 nhà a, b .
- Dòng tiếp theo là số nguyên dương k ($2 \leq k \leq n$) là số con giun ở Tpland.
- K dòng tiếp theo, mỗi dòng ghi 1 số nguyên dương d ($1 \leq d \leq n$) thể hiện ngôi nhà mà con giun thứ k đang sinh sống.

Kết quả: Ghi ra file văn bản **WORMS.OUT** một dòng duy nhất, ghi **NO** nếu cuộc gặp của giun không thể thực hiện được, nếu được, ghi một số nguyên duy nhất là thời gian để tất cả các con giun đi đến được điểm hẹn.

Ví dụ:

WORMS.INP	WORMS.OUT	
6 5 1 2 2 3 2 4 4 5 4 6 3 2 5 6	1	

BÀI 12. XE ĐỊA HÌNH

HSN và QT được quản lý một vùng đất rộng lớn trên mặt trăng nên họ muốn mua một chiếc ô tô địa hình để đi lại trên vùng đất này. Có thể hình dung vùng đất của HSN và QT như là lưới ô vuông $N \times N$ độ cao mỗi ô là như nhau và được biểu diễn bởi một số nguyên không âm. Xe địa hình đi lại trên mặt trăng đất hay rẻ tùy thuộc vào việc nó có thể đi giữa hai nơi có độ cao chênh lệch nhau nhiều hay ít. Giá của một chiếc xe địa hình có thể đi qua các ô có chênh lệch độ cao D đúng bằng D VND (đương nhiên là đi qua các ô có chênh lệch độ cao D thì cũng đi qua các ô có chênh lệch độ cao $< D$)

Yêu cầu: Hãy tính xem HSN và QT phải mua chiếc xe với giá tối thiểu bao nhiêu để có thể với chiếc xe này đi qua tối thiểu một nửa số ô của vùng đất. Nếu con số một nửa số ô vuông là con số lẻ thì nó được làm tròn đến số nguyên ở trên.

Dữ liệu: Vào từ file văn bản **XEDH.INP**

- Dòng đầu tiên ghi số nguyên dương $N \leq 500$
- N dòng tiếp theo, mỗi dòng chứa N số nguyên không âm mô tả độ cao của các ô vuông trong hàng tương ứng (giá trị không vượt quá 10^6)

Kết quả: Ghi ra file văn bản **XEDH.OUT** một số nguyên duy nhất là kết quả tìm được

Ví dụ

XEDH.INP	XEDH.OUT	Giải thích
5 0 0 0 3 3 0 0 0 0 3 0 9 9 3 3 9 9 9 3 3 9 9 9 9 3	3	Mua xe với giá là 3 có thể đi hết các ô có màu xám trên hình vẽ

BÀI 13. DFS

Cho đồ thị vô hướng có n đỉnh đánh số từ 1 đến n và m cạnh.

Hãy đếm số thành phần liên thông?

Dữ liệu: Vào từ file văn bản **DFS.INP**

- Dòng đầu tiên ghi hai số nguyên dương N, M ($N \leq 100000, M \leq 100000$)
- M dòng tiếp theo, mỗi dòng hai số nguyên dương u, v mô tả cạnh $u - v$

Kết quả: Ghi ra file văn bản **DFS.OUT** một số nguyên duy nhất là số thành phần liên thông tìm được.

BÀI 14. CÁC BĂNG ĐẢNG Ở CHICAGO

Năm 1920, thành phố Chicago bị các băng đảng xã hội đen lộng hành. Nơi đây, hàng nghìn gangster thường xuyên có mặt. Chúng luôn đối xử với nhau bằng luật giang hồ. Chúng có thể sẵn sàng hy sinh vì bạn mình hay sẵn sàng tiêu diệt những kẻ đã làm hại bạn bè chúng. Một nguyên tắc ở đây là:

- Bạn của bạn mình thì cũng là bạn mình.
- Kẻ thù của kẻ thù mình thì cũng là bạn mình.

Nếu 2 gangster là bạn của nhau, chắc chắn chúng ở cùng trong một băng đảng. Và các gangster trong một băng đảng thì chắc chắn là bạn của nhau.

Cảnh sát trưởng BINGO biết một số mối quan hệ giữa các tên gangster. Ông muốn dựa vào các mối quan hệ đó tính xem số lượng băng đảng lớn nhất là bao nhiêu?

Dữ liệu: Vào từ file văn bản **GANGSTER.INP**

- Dòng thứ nhất ghi số N là số các tên gangster. Các tên gangster được đánh số từ 1 đến N ($2 \leq N \leq 1000$).
- Dòng thứ hai ghi số M là số các mối quan hệ giữa các tên gangster mà BINGO biết ($1 \leq M \leq 5000$).
- M dòng tiếp theo, mỗi dòng ghi một mối quan hệ mà cảnh sát trưởng BINGO đã biết. Các mối quan hệ được ghi dưới dạng “E p q” hoặc “F p q”. Trong đó “E p q” thể hiện tên gangster p và tên gangster q là kẻ thù của nhau, “F p q” thể hiện tên gangster p và tên gangster q là bạn của nhau.

Kết quả: Ghi ra file văn bản **GANGSTER.OUT** một số duy nhất là số lượng băng đảng lớn nhất có thể.

Ví dụ:

GANGSTER.INP	GANGSTER.OUT
6 4 E 1 4 F 5 3 F 4 6 E 1 2	3 <u>Giải thích:</u> có 3 nhóm gangster {1}, {2, 4, 6}, {3, 5}.

BÀI 15. TALLER

Tại lớp học 12 chuyên tin, bạn lớp trưởng muốn kiểm tra chiều cao của các bạn trong lớp thông qua một bài toán thực tế như sau:

Biết:

- Lớp học có N học sinh được đánh số từ 1 đến N

- Có M thông tin được cung cấp, trong đó mỗi thông tin sẽ là một cặp số (x, y) thể hiện học sinh có chỉ số x cao hơn học sinh có chỉ số y . Ví dụ cặp $(4, 5)$, được xác định học sinh 4 cao hơn học sinh 5.

Yêu cầu: Các bạn hãy giúp lớp trưởng so sánh chiều cao của 2 bạn học sinh có chỉ số p và q bất kỳ xem ai cao hơn ai.

Dữ liệu: vào từ file văn bản **TALLER.INP** có dạng:

- Dòng 1: Hai số nguyên N và M , tương ứng là số học sinh trong lớp và số thông tin so sánh chiều cao của M cặp học sinh trong lớp. ($1 \leq N \leq 1.000.000$; $1 \leq M \leq 10.000.000$)
- M dòng tiếp theo, mỗi dòng ghi 2 số x, y tương ứng học sinh có chỉ số x cao hơn học sinh có chỉ số y . ($1 \leq x, y \leq N$)
- Dòng cuối ghi 2 số p và q tương ứng là câu hỏi so sánh chiều cao của học sinh p so với học sinh q xem ai cao hơn ($1 \leq p, q \leq N$)

Kết quả: Đưa ra file văn bản **TALLER.OUT** có dạng:

- Ghi **yes** nếu học sinh ở vị trí thứ p cao hơn học sinh ở vị trí thứ q
- Ghi **no** nếu học sinh ở vị trí thứ q cao hơn học sinh ở vị trí thứ p
- Ghi **unknown** trong trường hợp không so sánh được chiều cao của p và q .

Ví dụ:

TALLER.INP	TALLER.OUT
10 3	yes
8 4	
3 8	
4 2	
3 2	

TALLER.INP	TALLER.OUT
10 3	unknown
3 8	
2 8	
3 4	
3 2	

BÀI 16. ĐI BỘ NGẮM CẢNH

Tại thành phố HP có n điểm tham quan đánh số từ 1 đến n , điểm tham quan thứ i có chiều cao h_i và có m con đường một chiều nối n thành phố này.

Bạn hãy giúp HD tìm một con đường ngoạn mục là con đường có chênh lệch chiều cao là lớn nhất tức là cần chọn điểm bắt đầu s và điểm kết thúc f sao cho $h_f - h_s$ là lớn nhất.

Dữ liệu: Vào từ file văn bản **WALK.INP**

- Dòng 1: Ghi số nguyên dương t ($1 \leq t \leq 10$) là số test.
- Mỗi test được bắt đầu:
 - Dòng đầu tiên số nguyên dương n, n
 - m dòng tiếp theo, mỗi dòng ghi hai số nguyên dương a, b tương ứng có con đường 1 chiều đi từ a đến b

Kết quả: Ghi ra file văn bản **WALK.OUT** trên t dòng, dòng thứ i ghi kết quả của test i là giá trị lớn nhất của $h_f - h_s$

Ví dụ:

WALK.INP	WALK.OUT	
1 4 5 2 4 10 7 2 4 1 2 3 1 3 4 4 1	5	Tuyến đường: 1 → 2 → 4

Chú ý:

- Subtask 1 (20% số điểm): $1 \leq n \leq 1000, 1 \leq m \leq 2000$
- Subtask 2 (80% số điểm): $1 \leq n \leq 10^5, 1 \leq m \leq 2 \cdot 10^5$

BÀI 17. THREE MUSKETEERS

HD muốn chọn ra 3 chàng lính ngự lâm thỏa mãn một số tiêu chí:

- 3 chàng lính ngự lâm phải là bạn của nhau
- Tổng số lượng bạn của từng chàng lính (trừ 2 người cùng trong đội lính ngự lâm) là nhỏ nhất.

Dữ liệu: vào từ file văn bản **MUSKETEERS.INP** có dạng:

- Dòng 1: Hai số nguyên N và M , tương ứng là số lính và số thông tin bạn bè. ($3 \leq N \leq 4.000; 0 \leq M \leq 4.000$)
- M dòng tiếp theo, mỗi dòng ghi 2 số x, y tương ứng lính có chỉ số x là bạn của lính có chỉ số y . ($1 \leq x, y \leq N$)

Kết quả: Đưa ra file văn bản **MUSKETEERS.OUT** một số duy nhất là tổng số bạn nhỏ nhất của đội lính ngự lâm

Ví dụ:

MUSKETEERS.INP	MUSKETEERS.OUT
5 6 1 2 1 3 2 3 2 4 3 4 4 5	2

BÀI 18. NHỮNG CHÚ LÍNH CHÌ DỪNG CẢM

Có $2n$ chú lính chì xếp thành một hàng đôi. Họ có thể đổi chỗ cho nhau sao cho không có những chú lính chì có chiều cao bằng nhau ở mỗi hàng, khi đó chúng ta nói những chú lính chì đã xếp hàng đúng.

Một thao tác sẽ đổi cho 2 chú lính chì ở vị trí tương tự (ở các hàng khác nhau). Bạn hãy cho biết số thao tác nhỏ nhất để các chú lính chì xếp hàng đúng?

Ví dụ: Có 18 chú lính chì xếp thành 2 hàng như hình vẽ. Đường nối đôi chỉ sự thay đổi vị trí của 2 chú lính chì.

2	5	5	2	7	4	7	3	9
1	6	8	4	6	3	9	1	8

Dữ liệu: Vào từ file văn bản **DOUBLE.INP**

- Dòng 1: Số nguyên dương N ($1 \leq N \leq 50.000$)
- 2 dòng tiếp theo, mỗi dòng n số nguyên dương, số x_i ($1 \leq x_i \leq 100.000$) là chiều cao của chú lính thứ i của hàng.
- Dữ liệu vào luôn đảm bảo các chú lính chì có thể xếp hàng đúng.

Kết quả: Ghi ra file văn bản **DOUBLE.OUT** một số nguyên dương duy nhất là số thao tác đổi chỗ ít nhất để các chú lính chì xếp hàng đúng.

Ví dụ:

DOUBLE.INP	DOUBLE.OUT
9 2 5 5 2 7 4 7 3 9 1 6 8 4 6 3 9 1 8	3

BÀI 19. NHỮNG CHÚ LÍNH CHÌ 2015

Có $2n$ chú lính chì có chiều cao $(1, 1, 2, 2, 3, 3, \dots, n, n)$ xếp thành một hàng đôi. Một thao tác sẽ đổi cho 2 chú lính chì ở vị trí tương tự (ở các hàng khác nhau).

Một kiểu xếp hàng gọi là hoàn hảo nếu mỗi hàng là một hoán vị của N ?

Ví dụ:

1	5	5	3	4
3	2	2	4	1
Bình thường				

1	2	5	3	4
3	5	2	4	1
Hoàn hảo				

3	2	5	4	1
1	5	2	3	4
Hoàn hảo				

Yêu cầu: Cho một hàng đôi của các chú lính chì, hãy tìm:

- Số cách xếp hàng hoàn hảo có thể được sau một số phép thay đổi
- Số phép thay đổi ít nhất để được 1 kiểu xếp hàng hoàn hảo

Dữ liệu: Vào từ file văn bản **DOUBLE15.INP**

- Dòng 1 chứa số nguyên dương n
- 2 dòng tiếp theo, mỗi dòng n số nguyên dương, số x_i ($1 \leq x_i \leq n$) là chiều cao của chú lính thứ i của hàng.

Kết quả: Ghi ra file văn bản **DOUBLE15.OUT**

- Ghi hai số nguyên dương duy nhất theo thứ tự là số cách xếp hàng hoàn hảo có thể được sau một số phép thay đổi và số phép thay đổi ít nhất để được 1 kiểu xếp hàng hoàn hảo.

Các số trên một dòng của input file được ghi cách nhau bởi ít nhất 1 dấu cách

Ví dụ:

DOUBLE15.INP	DOUBLE15.OUT
5 1 5 5 3 4 3 2 2 4 1	4 1

Chú ý:

- Kết quả không vượt quá $2 \cdot 10^9$
- 40% số test tương ứng với 40% số điểm có $n \leq 20$
- 40% số test tương ứng với 40% số điểm có $20 < n \leq 400$
- 20% số test tương ứng với 20% số điểm có $400 < n \leq 10.000$

BÀI 20. BFS1

Cho đồ thị vô hướng N đỉnh, M cạnh (đồ thị thưa). Loại bỏ tất cả các cạnh đã cho và xây dựng các cạnh mới. Cho biết đường đi ngắn nhất từ S đến các đỉnh còn lại.

Dữ liệu: vào từ file văn bản **BFS1.INP**

- Dòng 1 số test t ($1 \leq t \leq 10$)
- Dòng 1: N, M ($2 \leq N \leq 2 \cdot 10^5, 1 \leq M \leq 120\,000$)
- M dòng tiếp theo, mỗi dòng 2 số nguyên u, v
- Dòng cuối S

Kết quả: ghi ra file văn bản **BFS1.OUT** $n - 1$ số nguyên trên một dòng theo thứ tự là độ dài từ S đến $1 \dots N$

Ví dụ:

BFS1.INP	BFS1.OUT
1 4 2 1 2 1 3 1	2 2 1

BÀI 21. ĐƯỜNG ĐI DÀI NHẤT CHIA HẾT CHO K

Cho cây có n đỉnh, $n - 1$ cạnh và một số nguyên dương k . Mỗi đỉnh của cây có một nhãn là a_i .

Yêu cầu: Bạn hãy tìm đường đi dài nhất trên cây sao cho mỗi đỉnh trên đường đi đều chia hết cho k .

Dữ liệu: Vào từ file văn bản **PATHK.INP**

- Dòng 1: Ghi số nguyên dương n ($1 \leq n \leq 10^5$) và số nguyên dương k ($1 \leq k \leq 10^5$)
- Dòng 2: n số nguyên dương a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^5$)
- $n - 1$ dòng tiếp theo, mỗi dòng hai số u, v mô tả có cạnh nối hai đỉnh $u - v$

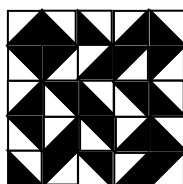
Kết quả: Ghi ra file văn bản **PATHK.OUT** một số nguyên duy nhất là độ dài đường đi dài nhất tìm được. Độ dài đường đi bằng số cạnh đi qua.

Ví dụ:

PATHK.INP	PATHK.OUT	
6 2 4 2 3 2 3 5 1 2 1 3 2 4 2 5 3 6	2	Tuyến đường: 1 \rightarrow 2 \rightarrow 4

BÀI 22. ĐẾM HÌNH

Có một loại gạch hình vuông, mỗi viên được phân đôi theo đường chéo, một nửa màu đen, còn nửa kia màu trắng. Người ta lát một diện tích hình vuông kích thước $N \times N$ bởi loại gạch nói trên. Các hình được tạo bởi các tam giác cùng màu kề cạnh và xung quanh có màu khác. Ví dụ, trên hình 1 có 10 hình màu đen, trong đó có 2 hình vuông, 2 hình bình hành, 1 hình thang, 4 hình tam giác và 2 hình khác.



Số 0



Số 1



Số 2



Số 3

Biểu diễn diện tích lát gạch bởi một mảng $A[i][j]$ ($i = 1, \dots, N; j = 1, \dots, N$ các hàng được đánh số từ trên xuống dưới bắt đầu từ 1, các cột được đánh số từ trái qua phải bắt đầu từ 1) trong đó $A[i][j]$ có giá trị là 0,1,2,3 nếu viên gạch ở hàng i , cột j có màu đen tương ứng nằm ở nửa trên đường chéo chính, nửa dưới đường chéo chính, nửa

trên đường chéo phụ, nửa dưới đường chéo phụ (hình 2). Chẳng hạn ma trận A biểu diễn sân ở hình 1 là:

```
3 1 1 3 1
0 2 3 2 2
3 0 1 0 1
0 3 0 3 1
1 2 0 3 2
```

Hãy đếm số lượng các hình màu đen có dạng hình tam giác, hình vuông, hình bình hành, hình thang và các hình khác.

Dữ liệu: Vào từ file văn bản **BRICK.INP**:

- Dòng đầu tiên ghi số nguyên dương N ($N \leq 50$)
- Dòng thứ i trong N dòng tiếp theo ghi N số $A[i][1], \dots, A[i][n]$. Các số trên một dòng cách nhau ít nhất một dấu cách.

Kết quả: Đưa ra file văn bản **BRICK.OUT** gồm 6 dòng lần lượt là: Số lượng các hình màu đen, số lượng các tam giác màu đen, số lượng các hình vuông màu đen, số lượng các hình bình hành màu đen, số lượng các hình thang màu đen, số lượng các hình khác màu đen.

Ví dụ:

BRICK.INP	BRICK.OUT
5	10
3 1 1 3 1	4
0 2 3 2 2	1
3 0 1 0 1	2
0 3 0 3 1	1
1 2 0 3 2	2

Chú ý: Bài này chấm từng dòng một.

BÀI 23. TRUYỀN TIN

Một lớp gồm N học sinh, mỗi học sinh cho biết những bạn mà học sinh đó có thể liên lạc được (chú ý liên lạc này là liên lạc một chiều : u có thể gửi tin tới v nhưng v thì chưa chắc đã có thể gửi tin tới u).

Thầy chủ nhiệm đang có một thông tin rất quan trọng cần thông báo tới tất cả các học sinh. Để tiết kiệm thời gian, thầy chỉ nhắn tin tới 1 số học sinh rồi sau đó nhờ các học sinh này nhắn lại cho tất cả các bạn mà các học sinh đó có thể liên lạc được, và cứ lần lượt như thế làm sao cho tất cả các học sinh trong lớp đều nhận được tin .

Hãy tìm một số ít nhất các học sinh mà thầy chủ nhiệm cần nhắn.

Dữ liệu: Vào từ file văn bản **MESSAGE.INP**

- Dòng đầu là N, M ($N \leq 800$, M là số lượng liên lạc 1 chiều)
- Một số dòng tiếp theo mỗi dòng gồm 2 số u, v cho biết học sinh u có thể gửi tin tới học sinh v .

Kết quả: Đưa ra file văn bản **MESSAGE.OUT** số học sinh cần thầy nhắn tin.

Ví dụ:

MESSAGE.INP	MESSAGE.OUT
12 15	2
1 3	
3 6	
6 1	
6 8	
8 12	
12 9	
9 6	
2 4	
4 5	
5 2	
4 6	
7 10	
10 11	
11 7	
10 9	

BÀI 24. BIẾN ĐỔI SỐ

Mỗi một số nguyên dương đều có thể biểu diễn dưới dạng tích của 2 số nguyên dương X, Y sao cho $X \leq Y$. Nếu như trong phân tích này ta thay X bởi $X - 1$ còn Y bởi $Y + 1$ thì sau khi tính tích của chúng ta thu được hoặc là một số nguyên dương mới hoặc là số 0.

Ví dụ: Số 12 có 3 cách phân tích $1*12, 3*4, 2*6$.

- Cách phân tích thứ nhất cho ta tích mới là 0: $(1 - 1) * (12 + 1) = 0$;
- cách phân tích thứ hai cho ta tích mới 10: $(3 - 1) * (4 + 1) = 10$;
- còn cách phân tích thứ ba cho ta 7: $(2 - 1) * (6 + 1) = 7$.

Nếu kết quả là khác 0 ta lại lặp lại thủ tục này đối với số thu được. Rõ ràng áp dụng liên tiếp thủ tục trên, cuối cùng ta sẽ đến được số 0, không phụ thuộc vào việc ta chọn cách phân tích nào để tiếp tục.

Yêu cầu: Cho trước số nguyên dương N ($1 \leq N \leq 100000$), hãy đưa ra tất cả các số nguyên dương khác nhau có thể gặp trong việc áp dụng thủ tục đã mô tả đối với N .

Dữ liệu: Vào từ file văn bản **WODAY.INP** gồm 1 dòng chứa số nguyên dương N .

Kết quả: Đưa ra file văn bản **WODAY.OUT** gồm 2 dòng:

- Dòng đầu tiên ghi K là số lượng số tìm được
- Dòng tiếp theo chứa K số tìm được theo thứ tự tăng dần bắt đầu từ số 0 .
- *Lưu ý: Có thể có số xuất hiện trên nhiều đường biến đổi khác nhau, nhưng nó chỉ được tính một lần trong kết quả.*

Ví dụ:

WODAY.INP	WODAY.OUT
12	6 0 3 4 6 7 10

Chú ý:

- Có 50% số test ứng với 50% số điểm của bài có $1 < N \leq 10^3$.
- Có 50% số test ứng với 50% số điểm của bài có $10^3 < N \leq 200\,000$.

BÀI 25. LÂU ĐÀI

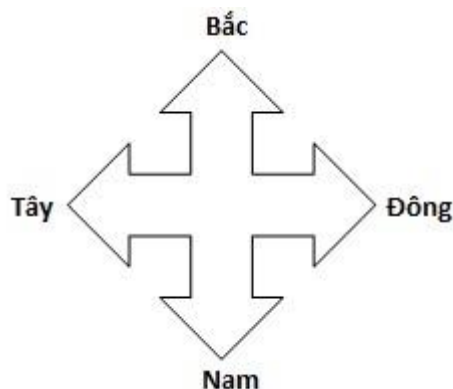
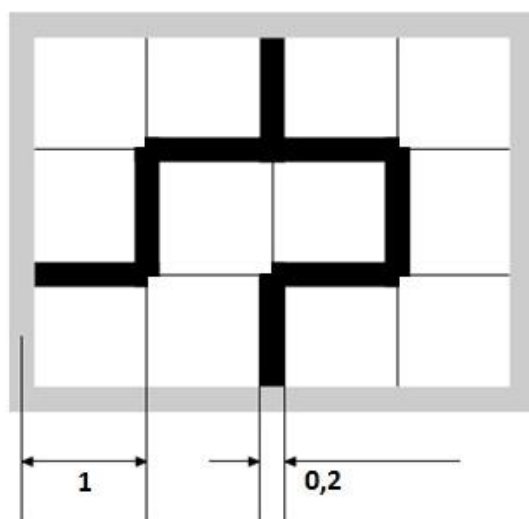
Một toà lâu đài hình chữ nhật kích thước $m \times n$ được chia thành lưới ô vuông đơn vị. Các dòng từ trên xuống được đánh số từ 1 tới m , các cột từ trái qua phải được đánh số từ 1 tới n . Quanh mỗi ô có thể có từ 0 tới 4 bức tường. Mỗi bức tường có chiều dài là 1 đơn vị và chiều rộng là 0,2 đơn vị. Quanh lâu đài có tường bao bọc. Để thể hiện tình trạng tường quanh một ô, ta gán cho mỗi ô một số nguyên, mà trong biểu diễn nhị phân của số nguyên đó:

- Bít 0 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Tây
- Bít 1 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Bắc
- Bít 2 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Đông
- Bít 3 bằng 1 hay 0 tùy theo ô đó có tường hay không có tường hướng Nam

Ví dụ trong hình vẽ dưới, ta có một lâu đài 3×4 .

Tình trạng tường của ô (2, 2) được thể hiện bởi số $3_{10} = 0011_2$

Tình trạng tường của ô (1,3) được thể hiện bởi số $11_{10} = 1011_2$



Lâu đài được chia thành các phòng, các phòng phân cách nhau bởi các bức tường. Hãy lập chương trình trả lời các câu hỏi sau:

1. Cho biết lâu đài có bao nhiêu phòng?
2. Cho biết diện tích của phòng rộng nhất?
3. Nếu được phá đi một và chỉ một bức tường thì được phòng có diện tích lớn nhất là bao nhiêu?

Dữ liệu: Vào từ file văn bản **CASTLE.INP**

- Dòng 1: Ghi hai số m, n ($1 \leq m, n \leq 100$);
- m dòng tiếp theo, dòng thứ i ghi n số nguyên, số thứ j thể hiện tình trạng tường quanh ô (i, j)

Dữ liệu vào bảo đảm tình trạng có tường tại các ô kề cạnh là không mâu thuẫn nhau và có ít nhất 2 phòng, các số trên một dòng được ghi cách nhau ít nhất một dấu cách.

Kết quả: Ghi ra file văn bản **CASTLE.OUT**

- Dòng 1: Ghi số phòng;
- Dòng 2: Ghi diện tích của phòng rộng nhất;
- Dòng 3: Ghi diện tích của phòng rộng nhất thu được sau khi phá tường.

(Kết quả câu 2 và câu 3 ghi theo quy cách gồm 2 chữ số sau dấu chấm thập phân)

Ví dụ:

CASTLE.INP	CASTLE.OUT
2 1	2
15	0.64
15	1.44

CASTLE.INP	CASTLE.OUT
3 4	3
3 14 11 6	3.86
13 3 14 5	7.10
11 12 11 12	

BÀI 26. MÊ CUNG

Một mê cung có dạng hình chữ nhật được chia thành lưới ô vuông gồm m hàng và n cột. Các hàng đánh số từ 1 đến m từ trên bắc xuống nam, các cột đánh số từ 1 đến n từ tây sang đông. Mỗi ô vuông được xem như một phòng, các cạnh của một ô vuông có tường hoặc không có tường. Người ta mô tả các bức tường của một ô vuông bởi một số nguyên nằm trong phạm vi từ 0 đến 15 bằng cách khởi đầu bởi giá trị 0, sau đó:

- Nếu có bức tường ở phía đông thì cộng thêm 1
- Nếu có bức tường ở phía bắc thì cộng thêm 2
- Nếu có bức tường ở phía tây thì cộng thêm 4
- Nếu có bức tường ở phía nam thì cộng thêm 8

Ví dụ: Nếu số mô tả là $10 = 2 + 8$ thì ô này có bức tường phía bắc và nam, còn phía đông và phía tây thì không có tường.

Một người xuất phát từ ô (x, y) - hàng x cột y với trong tay có k quả mìn cần di chuyển đến ô (u, v) - nơi đặt kho báu. Tại mỗi bước anh ta có thể di chuyển đến ô chung cạnh (với điều kiện ô này vẫn trong mê cung) và:

- Nếu không có tường, anh ta không mất quả mìn nào.
- Nếu có tường, anh ta mất 1 quả mìn (để phá tường). Tất nhiên điều này chỉ làm được nếu trên tay anh ta còn ít nhất một quả mìn.

Hãy tìm hành trình di chuyển với số bước di chuyển là nhỏ nhất?

Dữ liệu: Vào từ file văn bản **MAZE.INP:**

- Dòng đầu tiên là 3 số nguyên dương M, N và K ($1 \leq M, N \leq 100, 0 \leq K \leq 3$)
- Dòng thứ 2 ghi 4 số nguyên x, y, u, v

- M dòng sau, mỗi dòng gồm N số nguyên mô tả trạng thái mê cung

Kết quả: Ghi ra file văn bản **MAZE.OUT** một số nguyên duy nhất là số bước di chuyển tìm được (ghi -1 nếu như không có cách đi).

Ví dụ:

MAZE.INP	MAZE.OUT
3 4 1 1 1 3 3 7 6 2 3 5 5 4 1 12 9 12 9	4

BÀI 27. THOÁT KHỎI MÊ CUNG

Bạn Nghĩa bị kẹt trong một mê cung. Mê cung gồm $R \times C$ ô (R hàng, C cột). Có một số loại ô thuộc các loại khác nhau:

Loại ô	Ý nghĩa
#	tường
.	ô trống
*	vị trí của Nghĩa
B Y R G	ô xanh da trời, ô vàng, ô đỏ, ô xanh lá
b y r g	ô có khóa xanh da trời, vàng, đỏ, xanh lá
X	cửa thoát khỏi mê cung

Chú ý rằng mê cung có thể có nhiều cửa thoát hoặc không có cửa thoát, có thể có nhiều chìa khóa cùng màu, nhiều cửa cùng màu. Nghĩa được quyền đi theo 4 hướng. Nghĩa được quyền đi vào các ô trống và các ô có khóa. Với các ô màu, Nghĩa được quyền vào nếu trước đó Nghĩa đã đi qua ô có khóa màu tương ứng. Nghĩa không được quyền đi ra ngoài mê cung nếu đó không phải là ô X. Bạn hãy giúp Nghĩa tìm đường ngắn nhất đi ra khỏi mê cung.

Dữ liệu: vào từ file **MAZE1.INP** gồm nhiều bộ dữ liệu. Với mỗi bộ dữ liệu:

- Dòng thứ nhất là hai số R và C ($1 \leq R, C \leq 100$)
- R dòng tiếp theo, mỗi dòng ghi C kí tự thể hiện các ô trên một dòng của mê cung.
- Các bộ dữ liệu được ngăn cách bằng một dòng trống.
- File input kết thúc với $R = C = 0$.

Kết quả: ghi ra file **MAZE1.OUT**. Ứng với mỗi bộ dữ liệu, in ra một dòng:

- “*Escape possible in S steps.*” trong đó S là số bước ít nhất có thể thoát ra khỏi mê cung.
- “*The poor student is trapped!*” nếu không có cách thoát ra khỏi mê cung.

Ví dụ:

MAZE1.INP	MAZE1.OUT
<pre> 1 10 * X 1 3 * # X 3 20 ##### #XY.gBr.*.Rb.G.GG.y# ##### 0 0 </pre>	<pre> Escape possible in 9 steps. The poor student is trapped! Escape possible in 45 steps. </pre>

BÀI 28. KNIGHT07

Cho một bảng M hàng và N cột. Ô $(i, j) =$

- 0: ô chứa nước, mã nhảy vào thì chết đuối
- 1: ô trống, mã có thể nhảy vào
- 2: đá cuội, nhảy vào thì vỡ đầu chết
- 3: ô xuất phát
- 4: ô đích

Con mã có thể nhảy như sau:

- Nhảy theo một hướng nào đó M_1 ô
- Nhảy theo hướng vuông góc với hướng đã nhảy M_2 ô

Yêu cầu: Tìm số bước nhảy ít nhất từ ô xuất phát sang ô đích

Dữ liệu: Vào từ file văn bản **KNIGHT07.INP**

- Dòng 1: $M; N; M_1; M_2$ ($1 \leq M; N; M_1; M_2 \leq 30; M_1 \neq M_2$)
- M dòng tiếp theo, mỗi dòng N số $\in [0..4]$

Kết quả: Ghi ra file văn bản **KNIGHT07.OUT** một số duy nhất là số bước nhỏ nhất?

Ví dụ:

KNIGHT07.INP	KNIGHT07.OUT
<pre> 4 5 1 2 1 0 1 0 1 3 0 2 0 4 0 1 2 0 0 0 0 0 1 0 </pre>	<pre> 2 </pre>

BÀI 29. KNIGHT08

Cho một bảng M hàng và N cột. Ô $(i, j) =$

- 0: ô chứa nước, mã nhảy vào thì chết đuối
- 1: ô trống, mã có thể nhảy vào
- 2: đá cuội, nhảy vào thì vỡ đầu chết
- 3: ô xuất phát
- 4: ô đích

Con mã nhảy như luật cờ vua.

Yêu cầu:

- Cần phải biến đổi ít nhất bao nhiêu ô 0 thành 1 để mã nhảy từ ô xuất phát đến ô đích?
- Sau khi biến đổi ít nhất, số lượng bước nhảy ít nhất để đến đích là bao nhiêu?

Dữ liệu: Vào từ file văn bản **KNIGHT08.INP**

- Dòng 1: $M; N$ ($1 \leq M; N \leq 30$)
- M dòng tiếp theo, mỗi dòng N số $\in [0..4]$

Kết quả: Ghi ra file văn bản **KNIGHT08.OUT** trên 2 dòng tương ứng là đáp án 2 câu hỏi? Nếu không in 1 dòng -1

Ví dụ:

KNIGHT08.INP	KNIGHT08.OUT	
4 8	2	0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 1 0 0 0 0	6	0 x 0 0 0 2 0 1 0 0 0 0 0 2 0 1
0 0 0 0 0 2 0 1		0 0 0 0 x 4 0 0 0 0 x 0 x 4 0 0
0 0 0 0 0 4 0 0		3 0 0 0 0 0 1 0 3 0 0 0 0 0 1 0
3 0 0 0 0 0 1 0		

BÀI 30. ROADS

Có N thành phố và N-1 con đường hai chiều nối 2 thành phố bất kỳ và luôn tồn tại đường đi giữa 2 thành phố bất kỳ. Các con đường đã được xây dựng từ rất lâu và bây giờ tất cả các con đường cần sửa chữa. Người dân ở đây thực hiện ý định này bằng các con robot. Mỗi con robot sẽ sửa chữa con đường mà robot đang đứng, sau đó di chuyển sang con đường liên thuộc (chung một đỉnh) chưa được sửa chữa và sửa chữa nó.

Không có hai con robot cùng sửa một con đường và không có con đường nào được đi qua hai lần. Hãy tính số robot tối thiểu để hoàn thành công việc.

Dữ liệu: Vào từ file văn bản **ROADS.INP**

- Dòng 1: Số test T ($1 \leq t \leq 20$)
- Mỗi test bằng đầu:

- N số lượng thành phố, các thành phố đánh số từ 0 đến $N-1$ ($1 \leq n \leq 10000$)
- $n - 1$ dòng tiếp theo, dòng thứ i là hai số a_i, b_i ($0 \leq a_i, b_i < n$) tương ứng có cạnh nối hai đỉnh a_i, b_i

Kết quả: Ghi ra file văn bản **ROADS.OUT** trên t dòng, dòng thứ i là số robot ít nhất sử dụng trong test thứ i

Ví dụ:

ROADS.INP	ROADS.OUT	
3	1	Test 1: (0; 1) \rightarrow (0; 2) \rightarrow (0; 3)
4	1	Test 2: (0; 1) \rightarrow (1; 2) \rightarrow (2; 3) \rightarrow (2; 4) \rightarrow
0 1	2	(4; 5)
0 2		
0 3		
6		
0 1		
1 2		
2 3		
2 4		
4 5		
7		
0 1		
1 2		
2 3		
2 4		
4 5		
3 6		

BÀI 31. ROADS2

Cho đồ thị một chiều N đỉnh M cung. Đếm số lượng đường đi khác nhau từ 1 đến N .

Dữ liệu: Vào từ file văn bản **ROADS2.INP**

- Dòng 1: N, M ($1 \leq N \leq 10^4, 1 \leq M \leq 10^5$)
- M dòng tiếp theo, mỗi dòng 2 số nguyên u, v mô tả cung $u \rightarrow v$

Kết quả: Ghi ra file văn bản **ROADS2.OUT** số duy nhất là số lượng đường đi khác nhau từ 1 đến N theo modul 10^9 . Nếu có quá nhiều đường đi ghi **INFINITE PATHS**.

Ví dụ:

ROADS2.INP	ROADS2.OUT
5 5	2
1 2	
2 4	
2 3	
3 4	
4 5	

ROADS2.INP	ROADS2.OUT
5 5	INFINITE PATHS
1 2	
4 2	
2 3	
3 4	
4 5	

BÀI 32. SORTING

Cho dãy N số nguyên a_1, a_2, \dots, a_N trong đó $a_i \in [1..N], a_i \neq a_j \forall i \neq j$. Chi phí để đổi chỗ a_i, a_j là $a_i + a_j$.

Bạn hãy tính chi phí nhỏ nhất để sắp tăng dãy A .

Dữ liệu: Vào từ file văn bản **SORTING.INP**

- Dòng 1: N ($1 \leq N \leq 10.000$)
- N dòng tiếp theo, dòng i ghi $A[i]$

Kết quả: Ghi ra file văn bản **SORTING.OUT** một số nguyên duy nhất là chi phí nhỏ nhất?

Ví dụ:

SORTING.INP	SORTING.OUT	
3	7	- 2 3 1
2		- 2 1 3. Chi phí: 4
3		- 1 2 3. Chi phí 3
1		

BÀI 32. SỨC MẠNH PHI THƯỜNG

Cho một bảng vuông $n \times n$, mỗi ô trên bảng có thể trống hoặc chứa chương ngại vật. Hai ô gọi là kết nối trực tiếp với nhau nếu chung cạnh. Hai ô trống (r_1, c_1) và (r_2, c_2) gọi là liên thông với nhau nếu tồn tại một chuỗi các ô trống bắt đầu (r_1, c_1) , kết thúc (r_2, c_2) , và hai ô trống liên tiếp bất kì trong dãy có kết nối trực tiếp đến nhau.

BiBo có sức mạnh phi thường, anh ta có thể phá hủy được tất cả các chương ngại vật trong phạm vi hình vuông có kích thước $k \times k$, nhưng BiBo chỉ thực hiện việc phá hủy đó đúng một lần duy nhất.

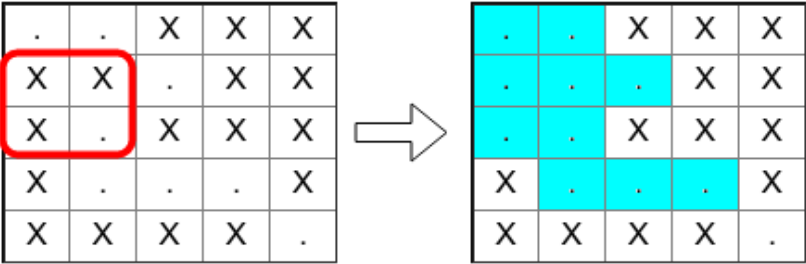
Yêu cầu: Hãy chọn hình vuông $k \times k$ trên lưới để sau khi BiBo phá hủy sẽ thu được một vùng có nhiều ô trống liên thông nhất có thể. Tính số lượng ô trống của vùng liên thông này.

Dữ liệu: Vào từ file văn bản **STRONG.INP**

- Dòng 1 chứa 2 số n và k ($1 \leq k \leq n \leq 500$) tương ứng là kích thước bảng và kích thước phạm vi BiBo có thể phá hủy trong một lần.
- Mỗi dòng trong n dòng sau chứa một xâu kí tự có độ dài n . mô tả lại trạng thái của bảng ban đầu với ô trống sẽ được kí hiệu là '.' nếu có chướng ngại vật được kí hiệu là 'X'.

Kết quả: Ghi ra file văn bản **STRONG.OUT** số lượng lớn nhất của vùng các ô trống liên thông với nhau sau khi được BiBo phá hủy các trường ngại vật trong phạm vi $k \times k$.

Ví dụ

STRONG.INP	STRONG.OUT	Giải thích
5 2 ..XXX XX.XX X.XXX X...X XXXX.	10	

BÀI 33. CHỊ HẰNG VÀ CHÚ CUỘI

Để chuẩn bị cho Lễ hội trăng rằm sắp diễn ra, công ty tổ chức sự kiện ABC cần chuẩn bị số lượng rất lớn các cặp đôi đóng vai chị Hằng và chú Cuội đến biểu diễn và trao quà cho các trường học cũng như các đơn vị cần tổ chức. Công ty hiện có n nhân viên, các nhân viên được đánh số từ 1 đến n , nhân viên thứ i sẽ có chỉ số về tài năng là i . Mỗi nhân viên sẽ có một *cấp trên trực tiếp*, có một người duy nhất không có cấp trên trực tiếp là giám đốc của công ty này. Nhân viên i được gọi là *cấp trên* của nhân viên j nếu i là cấp trên trực tiếp của j hoặc i là cấp trên của u và u là cấp trên trực tiếp của j .

Vì thời gian gấp rút trong khi số lượng hợp đồng rất nhiều, nên Ban giám đốc công ty muốn nhờ bạn đếm số cặp nhân viên có thể đóng vai chị Hằng và chú Cuội cùng nhau, biết để tạo thành một cặp đôi thì hai nhân viên i và j phải thỏa mãn 2 điều kiện sau:

- i là cấp trên của j .
- Chênh lệch về tài năng giữa hai nhân viên không vượt quá k , tức là $|i - j| \leq k$ để việc đóng cặp được thuận lợi

Yêu cầu: Bạn hãy giúp ban giám đốc của công ty ABC đếm số cặp đôi này.

Dữ liệu: Vào từ file văn bản **PAIR.INP**

- Dòng đầu tiên chứa 2 số nguyên n, k ($1 \leq n, k \leq 10^5$).
- $n - 1$ dòng tiếp theo, mỗi dòng chứa 2 số nguyên u, v ($1 \leq u, v \leq n$) miêu tả mối quan hệ nhân viên u là cấp trên trực tiếp của nhân viên v . Dữ liệu đảm bảo các mối quan hệ trong công ty tạo thành một cấu trúc cây.

Kết quả: Ghi ra file văn bản **PAIR.OUT** một số nguyên không âm duy nhất là kết quả của bài toán.

Ví dụ:

PAIR.INP	PAIR.OUT
5 2 3 2 3 1 1 4 1 5	4

Chú ý:

- Sub1: 30% số test có $n \leq 300$.
- Sub2: 30% số test tiếp theo có $n \leq 5000$

BÀI 34. KNIGHT09

Cho bàn cờ vua cỡ $N \times M$. Trên một số ô có sẵn 1 số quân mã (đôi một khác nhau). Hai con mã có thể đổi cho cho nhau nếu cả hai con đều đồng thời đến ô của nhau sau 1 bước đi.

Bạn hãy cho biết có tất cả bao nhiêu trạng thái của bàn cờ (theo modulo $10^9 + 7$)

Dữ liệu: Vào từ file văn bản **KNIGHT09.INP**

- Dòng đầu tiên: số test t ($1 \leq t \leq 10$)
- Mỗi test được bắt đầu bằng:
 - o Dòng 1: N, M, Q ($1 \leq N, M \leq 10^3, 1 \leq Q \leq N * M$) tương ứng là kích thước bàn cờ và số quân mã
 - o Q dòng tiếp theo, dòng thứ i ghi số nguyên dương x, y là tọa độ quân mã i

Kết quả: Ghi ra file văn bản **KNIGHT09.OUT** trên t dòng, mỗi dòng một số nguyên duy nhất là số trạng thái của bàn cờ?

Ví dụ:

KNIGHT09.INP	KNIGHT09.OUT	Giải thích
2 4 4 4 1 1 1 2 3 1 3 2 4 4 4 1 1 1 2 3 1 4 4	4 2	Test 1: - Quân mã 1 và 4 có thể đổi chỗ cho nhau - Quân mã 2 và 3 có thể đổi chỗ cho nhau. Vậy trên bàn cờ có thể có 4 trạng thái xếp quân cờ

Chú ý:

- Sub1: 20% số test có $1 \leq t \leq 10, 1 \leq n, m \leq 10$.
- Sub2: 30% số test tiếp theo có $1 \leq t \leq 10, 1 \leq n, m \leq 100$.

BÀI 35. BIẾN ĐỔI BẢNG

Cho một bảng $n \times m$ chứa các số 0, 1 và 2.

Một ô chứa số 1, kề cạnh với ô chứa số 2 thì có thể biến đổi từ 1 thành 2 mất 1 giây.

Hãy tính thời gian ngắn nhất để tất cả các ô số 1 thành số 2.

Dữ liệu: Vào từ file văn bản **BIENDO12.INP**

- Dòng 1 chứa 2 số n và m ($1 \leq n, m \leq 10^3$) tương ứng là kích thước bảng
- Mỗi dòng trong n dòng sau chứa m số nguyên 0, 1, 2 mô tả bảng

Kết quả: Ghi ra file văn bản **BIENDO12.OUT** một số nguyên duy nhất là thời gian ngắn nhất.

Ví dụ

BIENDO12.INP	BIENDO12.OUT
3 5	2
2 1 0 2 1	
1 0 1 2 1	
1 0 0 2 1	

BÀI 36. ĐỒ THỊ CÓ CẠNH BIẾN MẤT

Cho đồ thị vô hướng có N đỉnh (đánh số từ 1 đến N) và M cạnh (đánh số từ 1 đến M). BiBo đang ở đỉnh 1 và muốn di chuyển đến đỉnh N , biết rằng:

- Di chuyển trên 1 cạnh mất 1 đơn vị thời gian
- Thời điểm ban đầu $t = 0$ thì BiBo đang ở đỉnh 1
- Có K thời điểm, một số cạnh sẽ bị biến mất khỏi đồ thị trong 1 đơn vị thời gian (không thể di chuyển trên cạnh này).

Hãy tính thời gian ngắn nhất đi từ 1 đến N .

Dữ liệu: Vào từ file văn bản **DISAPPEAR.INP**

- Dòng đầu tiên chứa số nguyên n, m, k ($1 \leq n \leq 10^5, 0 \leq k \leq M \leq \min\left(\frac{N \times (N-1)}{2}; 10^5\right)$).
- m dòng tiếp theo, mỗi dòng chứa 2 số nguyên u, v ($1 \leq u, v \leq n$) mô tả cạnh
- k dòng tiếp theo, mỗi dòng hai số nguyên dương t, x ($1 \leq t \leq 10^9, 1 \leq x \leq m$) tương ứng ở thời điểm t thì cạnh chỉ số x biến mất khỏi đồ thị

Kết quả: Ghi ra file văn bản **DISAPPEAR.OUT** một số nguyên không âm duy nhất là kết quả của bài toán.

Ví dụ:

DISAPPEAR.INP	DISAPPEAR.OUT
5 5 1 1 2 1 3 2 5 3 4 4 5 1 3	3

BÀI 37. TRÒ CHƠI

Một chương trình trò chơi trên máy tính được mô tả như sau:

Cho 2 số X, Y và một dãy số nguyên dương a_1, a_2, \dots, a_n . Chương trình thực hiện các bước sau :

1. Ban đầu $X = 1$ và $Y = 0$. Sau một số bước, nếu $X \leq 0$ hoặc $X > n$ thì lập tức dừng chương trình.
2. Tăng cả X và Y lên một lượng là a_x
3. Giảm X đi một lượng là a_x và tăng Y lên một lượng là a_x
4. Chương trình thực hiện lần lượt bước 2 và bước 3 liên tục đến khi nào không thỏa mãn.

Yêu cầu: Cho dãy a_2, a_3, \dots, a_n . Giả sử với mỗi i ($1 \leq i \leq n - 1$), gán $a_1 = i$, chạy chương trình trên với dãy a_1, a_2, \dots, a_n . Với mỗi giá trị, in ra giá trị cuối cùng của Y , in -1 nếu chương trình không bao giờ kết thúc.

Dữ liệu: Vào từ file văn bản **GAMEC.INP**:

- Dòng đầu tiên cho số nguyên n ($2 \leq n \leq 2.10^5$)
- Dòng thứ 2 chứa $n - 1$ số nguyên a_2, a_3, \dots, a_n ($1 \leq a_i \leq 10^9$).

Kết quả: Ghi ra file văn bản **GAMEC.OUT** trên $n - 1$ dòng, dòng i ứng với kết quả của y khi gán $a_1 = i$.

Ví dụ :

GAMEC.INP	GAMEC.OUT	Giải thích
4	3	$i = 1: X = 1 \rightarrow 2 \rightarrow 0$. Khi đó $Y = 1 + 2 = 3$
2 4 1	6	$i = 2: X = 1 \rightarrow 3 \rightarrow -1$. Khi đó $Y = 2 + 4 = 6$
	8	$i = 3: X = 1 \rightarrow 4 \rightarrow 3 \rightarrow 7$. Khi đó $Y = 3 + 1 + 4 = 8$

Chú ý:

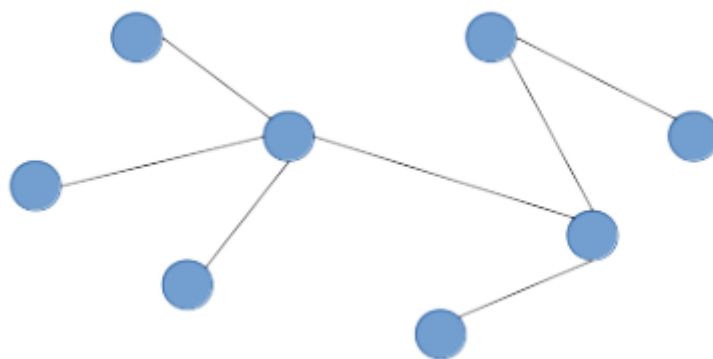
- 40 % số test có $n \leq 100, a_i \leq 1000$.
- 30 % số test có $n \leq 1000, a_i \leq 10^9$.

BÀI 38. PHỞ NAM ĐỊNH

@Bi@-@Bo@ là một blogger chuyên viết bài về ẩm thực. Hôm nay, anh ấy muốn thăm các nhà hàng Phở ở Nam Định để xem nhà hàng Phở nào ngon nhất.

Có n nhà hàng ở Nam Định, đánh số từ 0 đến $n - 1$. Tuy nhiên, chỉ có m nhà hàng trong số chúng là bán Phở. @Bi@-@Bo@ chọn một nhà hàng bất kì để bắt đầu tìm hiểu. Có $n - 1$ con đường ở Nam Định, mỗi con đường nối chính xác hai nhà hàng. Đảm bảo tất cả các nhà hàng được kết nối bởi các con đường này. Mỗi con đường @Bi@-@Bo@ mất 1 phút để đi qua nó.

Trong khoa học máy tính, một mạng các con đường với cấu trúc như trên được gọi là một cây. Như ví dụ trong hình sau:



Yêu cầu: Xác định tổng thời gian nhỏ nhất để @Bi@-@Bo@ đi trên các con đường để đến tất cả các nhà hàng Phở.

Dữ liệu: Vào từ file văn bản **PHO.INP**

- Dòng 1 chứa hai số nguyên n và m ($2 \leq m \leq n \leq 10^5$)
- Dòng 2 chứa m số nguyên khác nhau là số hiệu các nhà hàng Phở.
- $n - 1$ dòng tiếp theo, mỗi dòng hai số nguyên a và b ($0 \leq a, b \leq n - 1$) mô tả một con đường nối hai nhà hàng a và b .

Kết quả: Ghi ra file văn bản **PHO.OUT** một số duy nhất là tổng thời gian ít nhất @Bi@-@Bo@ đi qua các con đường thăm hết m nhà hàng Phở.

Ví dụ

PHO.INP	PHO.OUT
8 2	3
5 2	
0 1	
0 2	
2 3	
4 3	
6 1	
1 5	
7 3	

Chú ý:

- 30 % số test có $M = 2$ và $N \leq 100$.
- 30 % số test có $M \leq N \leq 10^4$.

BÀI 38. CHÈO THUYỀN

Người dân nước GeoLand say mê các môn thể thao mạo hiểm đòi hỏi tư duy hình học chuyên nghiệp. Một trong những môn thể thao đó là bơi thuyền vượt bãi đá trên sông Rect River – con sông dài nhất GeoLand. Bản đồ con sông được vẽ trên mặt phẳng tọa độ với hệ tọa độ descartes vuông góc, hai bờ sông là hai đường thẳng song song $y = 0$ và $y = h$. Bãi đá trên sông gồm n tảng đá đánh số từ 1 tới n , tảng đá thứ i có tọa độ (x_i, y_i) trên bản đồ. Mỗi vận động viên tham gia bài thi với một thuyền thúng hình tròn. Anh ta được đặt thuyền của mình ở vị trí tùy chọn nằm hoàn toàn bên trái bãi đá và cần bơi thuyền tới một vị trí tùy chọn nằm hoàn toàn bên phải bãi đá. Thuyền được di chuyển theo hướng tùy ý nhưng không được chạm vào bờ sông hay chạm vào một tảng đá nào của bãi đá (kể cả đường biên của thuyền).

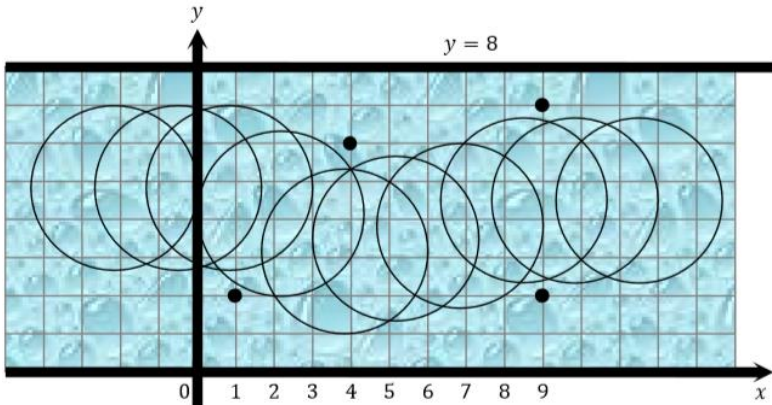
Yêu cầu: Tìm số nguyên d lớn nhất để mọi thuyền có đường kính $< d$ đều có thể thực hiện được bài thi.

Dữ liệu: Vào từ file văn bản **ROWING.INP**

- Dòng 1 chứa hai số nguyên dương n, h ($n \leq 4000; 2 \leq h \leq 10^9$)
- n dòng tiếp theo, mỗi dòng chứa hai số nguyên dương $x_i \leq 10^9, y_i < h$.

Kết quả: Ghi ra file văn bản **ROWING.OUT** một số nguyên duy nhất là số d tìm được.

Ví dụ:

ROWING.INP	ROWING.OUT	Giải thích
4 8 1 2 4 6 9 2 9 7	5	

BÀI 39. TRỜI CHO

Bi và Bo cùng nhau chơi 1 trò chơi dạng như tetris. Các khối hình được đặt lên nhau tạo thành hình khối có chiều dài B, chiều cao L và chiều rộng 1.

Tính chất đặc biệt của hình khối nay là các khối liền kề luôn tạo thành cụm.

Ví dụ:

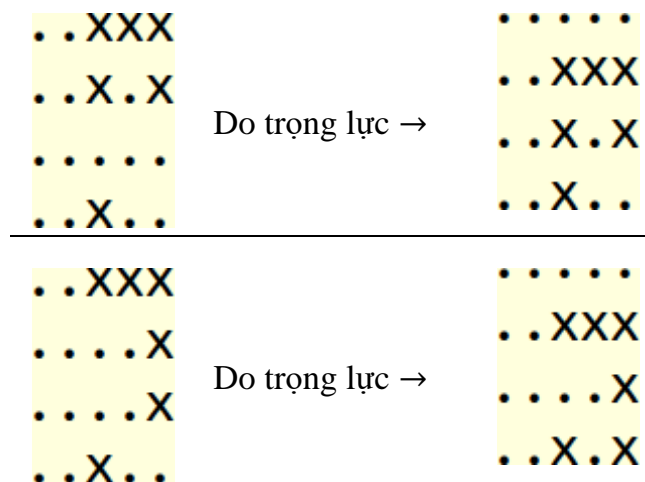
```

    . . X . .
    . . X . .
    X X X . .
    . . X . .
  
```

(Hình khối này có $L = 4, B = 5$, 'X' là biểu thị các hình khối nhỏ chiều rộng 1, '.' là không gian trống, các hàng đánh số từ 1 đến L từ dưới lên trên, các cột đánh số từ 1 đến B từ trái qua phải)

Biết rằng, để tạo thành một cụm ổn định thì các khối hình nhỏ phải dính liền vào nhau, nếu không, do tác dụng của trọng lực, các khối hình nhỏ sẽ rơi xuống.

Ví dụ:

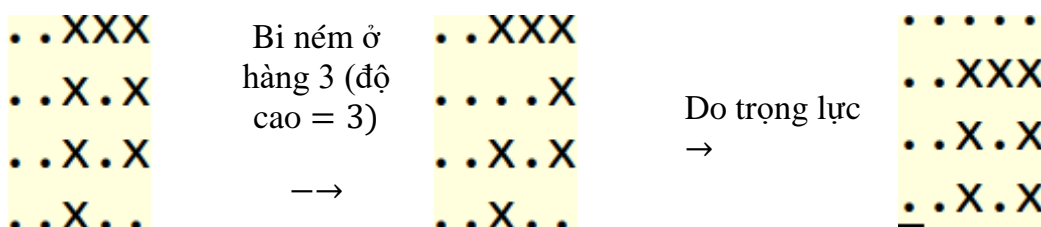


Như vậy, nếu các khối hình nhỏ rơi xuống, nếu tiếp xúc với đất hoặc khối hình khác thì sẽ dừng rơi xuống tiếp.

Trò chơi như sau:

- Bi và Bo sẽ thay nhau ném các phi tiêu (gắn đầu đạn hạt nhân). Bi ném đầu tiên
- Bi ném từ trái qua phải, gặp khối hình nhỏ đầu tiên thì cả khối hình nhỏ và phi tiêu đều nổ (tan biến trong hư vô)
- Bo ném từ phải qua trái, gặp khối hình nhỏ đầu tiên thì cả khối hình nhỏ và phi tiêu đều nổ (tan biến trong hư vô)

Ví dụ:



Yêu cầu: Hãy in ra trạng thái cuối cùng sau khi kết thúc trò chơi. Biết rằng:

- Tại thời điểm bất kỳ, không có hai cụm cùng rơi 1 lúc.
- Trạng thái ban đầu là ở trạng thái ổn định

Dữ liệu: Vào từ file văn bản **GAMEBIBO.INP**

- Dòng 1: L và B ($1 \leq L, B \leq 100$)
- L dòng tiếp theo, mỗi dòng B ký tự 'X' hoặc '.' mô tả trạng thái ban đầu
- Dòng tiếp theo chứa số N là số lần chơi của cả Bi và Bo ($1 \leq N \leq 100$)
- Dòng tiếp theo chứa N số nguyên theo thứ tự là dòng bị ném phi tiêu.

Kết quả: Ghi ra file văn bản **GAMEBIBO.OUT** trạng thái cuối cùng của trò chơi

Ví dụ:

GAMEBIBO.INP	GAMEBIBO.OUT
5 6
.....
..xx..	..xx..
..x...	..xx..
..xx..	..xxxx.
..xxxx.	
1	
3	

BÀI 40. CPATH

Đất nước CBN có n thành phố và m con đường một chiều phục vụ việc đi lại giữa các thành phố. Ông An, một khách du lịch VIP muốn xuất phát từ thành phố 1 đi qua đúng k con đường để tới thành phố n . Các con đường có thể đi lại nhiều lần.

Yêu cầu: Hãy xác định số hành trình thỏa mãn yêu cầu của ông An.

Dữ liệu: Vào từ file văn bản **CPATH.INP**:

- Dòng đầu tiên chứa 3 số nguyên n, m, k ($n \leq 50, m \leq 1000, k \leq 10^9$)
- m dòng tiếp theo, dòng thứ i chứa 2 số nguyên u_i, v_i xác định một con đường đi từ u_i tới v_i . Giữa hai thành phố có thể có nhiều con đường nối trực tiếp.

Kết quả: Đưa ra file văn bản **CPATH.OUT** một số nguyên duy nhất là số lượng hành trình đường thỏa mãn đưa ra theo *module* 2017.

Ví dụ:

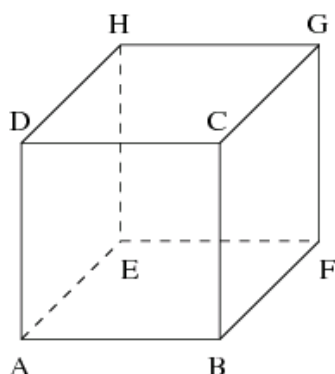
CPATH.INP	CPATH.OUT
3 4 4	2
1 2	
2 1	
2 3	
3 2	

Ràng buộc:

- 30% số test tương ứng 30% số điểm có $n \leq 5, m \leq 10, k \leq 6$
- 30% số test khác tương ứng 30% số điểm có $n \leq 50, k \leq 1000$

BÀI 41. ANT

Một con kiến đang muốn đi dọc theo các cạnh của hình hộp chữ nhật ABCDEFGH



Con kiến này muốn tính xem có bao nhiêu con đường có thể di chuyển từ 1 đỉnh tới một đỉnh khác qua k cạnh (khi một con kiến đi vào một cạnh, nó không thể quay trở lại và phải kết thúc ở đỉnh cuối của cạnh này). Nếu con kiến đi qua một vài cạnh x lần, chúng ta đếm những cạnh này được đi qua x lần. Những con kiến muốn có một tuyến đường đi thú vị, chúng không rời khỏi một đỉnh bằng cạnh mà chúng đã vào đỉnh đó (Nó không muốn sử dụng cùng một cạnh trong một dãy bước di chuyển liên nhau).

Con kiến của chúng ta không được thông minh, bởi vậy nó chỉ sử dụng được các số nguyên từ 0 đến $p - 1$. Vì vậy, bạn hãy tìm kết quả theo modul P .

Dữ liệu: Vào từ file văn bản **ANT.INP**

- Dòng 1: 2 ký tự in hoa V_1 và V_2 ($A \leq V_1; V_2 \leq H$) ($V_1 \neq V_2$). Hai ký tự này lần lượt là điểm xuất phát và điểm kết thúc của cuộc hành trình.
- Dòng 2: 2 số nguyên K và P ($1 \leq k \leq 2.000.000.000$), ($2 \leq p \leq 1.000.000.000$)

Kết quả: Ghi ra file văn bản **ANT.OUT** một số nguyên duy nhất là số tuyến đường đi thú vị từ V_1 đến V_2 đi qua k cạnh và theo modul P

Ví dụ:

ANT.INP	ANT.OUT	
A B 3 100	2	

BÀI 42. DIJ

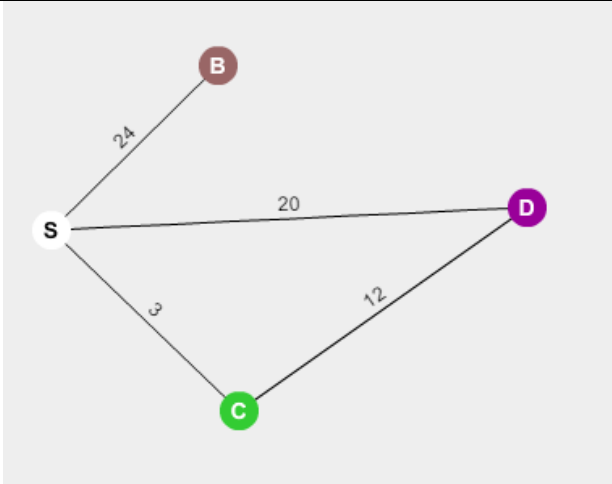
Cho đồ thị N đỉnh có trọng số. Tìm đường đi ngắn nhất từ S đến các đỉnh khác S . Nếu không có đường đi in -1.

Dữ liệu: Vào từ file văn bản **DIJ.INP**

- Dòng 1: Số test t ($1 \leq t \leq 10$)
 - o Dòng 1: N, M ($2 \leq N \leq 3000, 1 \leq m \leq n \times \frac{n-1}{2}$)
 - o M dòng tiếp theo, mỗi dòng u, v, c ($1 \leq u, v \leq n; 1 \leq c \leq 10^5$). Mỗi cặp đỉnh u, v có thể có nhiều cạnh nối.
 - o Dòng tiếp theo: s

Kết quả: Ghi ra file văn bản **DIJ.OUT** trên nhiều dòng, mỗi dòng ghi $n - 1$ số nguyên là đường đi ngắn nhất từ S tới mọi đỉnh

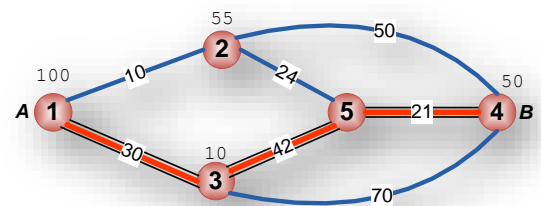
Ví dụ:

DIJ.INP	DIJ.OUT	
1 4 4 1 2 24 1 4 20 3 1 3 4 3 12 1	24 3 15	

BÀI 43. ĐỒNG BẠC CỎ

Rôn là một người cởi mở và vì vậy có rất nhiều bạn bè. Một buổi tối, khi mở Mail, Rôn ngạc nhiên một cách thú vị khi thấy Mail của một người bạn cũ thời niên thiếu mời tới họp mặt. Không một chút lưỡng lự, Rôn nhận lời. Biết rằng bạn mình say mê sưu tập tiền cỏ và trong bộ sưu tập còn thiếu một đồng bạc đặc biệt thời trung cổ.

Nước của Rôn có n thành phố, đánh số từ 1 đến n nối với nhau bởi m đường hai chiều, mỗi con đường nối một cặp hai thành phố khác nhau và mỗi cặp thành phố có không quá một con đường nối trực tiếp. Rôn ở thành phố **A**, người bạn - ở thành phố **B** ($A \neq B$). Qua thông tin mà bạn bè



và internet cung cấp Rôn biết danh sách k thành phố có bán đồng tiền này và giá bán ở mỗi thành phố. Internet cũng cho biết chi phí d_{ij} đi từ i tới j nếu hai thành phố này có đường nối trực tiếp.

Rôn quyết định sẽ lái xe đi từ A tới B và sẽ mua đồng tiền cổ ở một trong số các thành phố trên đường đi. Vấn đề là phải chọn đường đi sao cho tổng chi phí đi cộng với chi phí mua đồng tiền cổ là nhỏ nhất.

Yêu cầu: Xác định tổng chi phí nhỏ nhất để thực hiện kế hoạch của Rôn.

Dữ liệu: Vào từ file văn bản **MONEY.INP**:

- Dòng đầu tiên chứa 3 số nguyên n, m và k ($2 \leq n \leq 5000, 1 \leq m \leq 100\,000, 1 \leq k \leq n$);
- Dòng thứ 2 chứa 2 số nguyên A và B ;
- Dòng thứ 3 chứa k cặp số nguyên, mỗi cặp xác định thành phố và giá bán đồng tiền cổ (nằm trong phạm vi từ 1 đến 10^9), ở các thành phố khác nhau – giá khác nhau;
- Mỗi dòng trong m dòng còn lại chứa 3 số nguyên i, j và d_{ij} ($1 \leq d_{ij} \leq 10^5$).

Kết quả: Đưa ra file văn bản **MONEY.OUT** một số nguyên – chi phí nhỏ nhất tìm được.

Ví dụ:

MONEY.INP	MONEY.OUT
5 7 4	103
1 4	
1 100 4 50 3 10 2	
55	
1 2 10	
5 3 42	
1 3 30	
2 4 50	
3 4 70	
2 5 24	
4 5 21	

BÀI 44. PENCIL

Tại HT land có tất cả N thành phố, mỗi thành phố có những tuyến đường đi khác nhau đến thành phố khác. Tất cả có T con đường tại HT land. Với mỗi con đường đi giữa hai thành phố x và y có một lệ phí chuyên chở là $C(x, y)$. Trong N thành phố này, có K thành phố có bán những chiếc bút chì rất đẹp và có thể mua trực tiếp. Giá của 1 chiếc bút chì ở thành phố x là P_x .

HT muốn xuất phát từ thành phố D mua 1 chiếc bút chì với giá nhỏ nhất (đương nhiên bao gồm giá bút chì + phí đi đường). Bạn hãy giúp HT giải quyết vấn đề trên?

(Chú ý: Có thể mua bút chì ngay tại thành phố D mà không cần phải di chuyển sang thành phố khác)

Dữ liệu: Vào từ file văn bản **PENCIL.INP**

- Dòng 1: Số nguyên dương N là số thành phố ở HT land. Các thành phố được đánh số từ 1 đến N . ($N \leq 5\,000$)
- Dòng 2: Số T là số con đường ở HT land. ($0 \leq T \leq 500\,000$)
- T dòng tiếp theo, mỗi dòng có 3 số nguyên $x, y, C(x, y)$ có nghĩa là trên con đường nối x và y (y và x) có chi phí là $C(x, y)$. ($0 \leq C(x, y) \leq 10\,000$)
- Dòng tiếp theo là số nguyên K , là số thành phố ở HT land có bán bút chì.
- K dòng tiếp theo chứa 2 số nguyên z và P_z có nghĩa là 1 chiếc bút chì ở thành phố z bán với giá là P_z ($0 \leq p_x \leq 10\,000$).
- Dòng cuối là số nguyên D , là thành phố HT xuất phát.

Kết quả: Đưa ra file văn bản **PENCIL.OUT** một số nguyên là số chi phí nhỏ nhất để HT mua được một chiếc bút chì?

Ví dụ:

PENCIL.INP	PENCIL.OUT
3	6
3	
1 2 4	
2 3 2	
1 3 3	
3	
1 14	
2 8	
3 3	
1	

BÀI 45. INCPATH

Mỗi sáng HD và các bạn hay đi bộ qua các khu phố. Có n ($n \leq 250$) khu phố nối với nhau bởi m ($m \leq 25,000$) con đường hai chiều. Mỗi con đường i có độ dài l_i . Họ sẽ luôn đi hành trình ngắn nhất từ phố 1 đến phố n .

HD đã chọn 1 con đường và nhân đôi độ dài con đường đó lên nhằm mục đích khiến cho hành trình của họ trở nên dài nhất có thể (??). Nhiệm vụ của bạn là giúp HD tính xem độ dài tối đa tăng thêm là bao nhiêu.

Độ dài các con đường không lớn hơn 10^6 .

Dữ liệu: Vào từ file văn bản **INCPATH.INP**

- Dòng 1: N, M ($2 \leq N \leq 250, 1 \leq m \leq 25000$)
- M dòng tiếp theo, mỗi dòng u, v, c ($1 \leq u, v \leq n; 1 \leq c \leq 10^6$).

Kết quả: Đưa ra file văn bản **INCPATH.OUT** một số nguyên – độ dài tối đa tăng thêm

Ví dụ:

INCPATH.INP	INCPATH.OUT
5 7 2 1 5 1 3 1 3 2 8 3 5 7 3 4 3 2 4 7 4 5 2	2 {chọn đường số 5}

BÀI 46. BIN LADEN

Trùm khủng bố Bin Laden trốn trong 1 căn hầm được đào sâu xuống mặt đất M tầng, mỗi tầng có N phòng. Các phòng được ngăn cách bằng các cửa rất khó phá. Các phòng có cửa xuống phòng ngay phía dưới và 2 phòng ở 2 bên. Từ trên mặt đất có N cửa xuống N phòng tầng -1 . Bin Laden ở tầng dưới cùng (tầng $-M$) phòng thứ N (phòng ở bên phải nhất). Mỗi cửa được làm bằng một kim loại khác nhau với độ dày khác nhau nên việc phá cửa cần thời gian khác nhau.

Bạn hãy tìm cách đi từ mặt đất xuống phòng của Bin Laden nhanh nhất.

Dữ liệu: Vào từ file văn bản **BINLADEN.INP**

- Dòng 1 ghi M và $N * 1 \leq M \leq 2222, 1 \leq N \leq 10$
- Dòng 2 đến $2M + 1$, dòng chẵn ghi N số, dòng lẻ ghi $N - 1$ số là chi phí để phá cửa. Chi phí của các cánh cửa thuộc $[0, 1000]$

Kết quả: Đưa ra file văn bản **BINLADEN.OUT** một số là thời gian nhỏ nhất để đến được phòng của Bin Laden

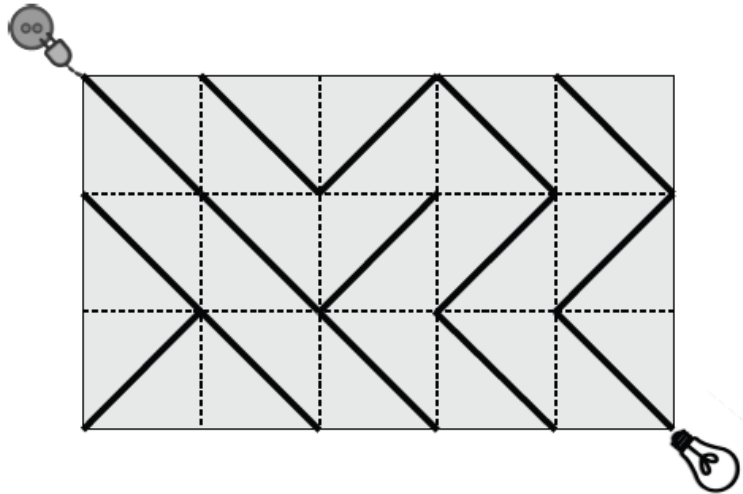
Ví dụ

BINLADEN.INP	BINLADEN.OUT	Giải thích
4 2 99 10 1 10 99 1 99 10 1 10 99 1	44	<pre> +--99--+--10--+ 1 +--10--+--99--+ 1 +--99--+--10--+ 1 +--10--+--99--+ 1 +-----+-----+ Đi theo đường zigzac </pre>

BÀI 46. SWITCH THE LAMP ON

HD phát minh ra một bảng điện có dạng hình chữ nhật $N \times M$. Tại mỗi hình vuông đơn vị, 2 trong 4 góc được nối với nhau bởi dây điện.

Nguồn điện được đặt tại góc trên bên trái của bảng điện, bóng đèn được đặt ở góc dưới bên phải (như hình vẽ)



Trong hình vẽ này, bóng đèn bị tắt. Nhưng nếu bất kỳ đoạn dây điện nào ở cột 4 quay 90° thì bóng đèn sẽ sáng.

Yêu cầu: Hãy tính số đoạn dây cần quay 90° ít nhất để bóng đèn sáng.

Dữ liệu: Vào từ file văn bản **LAMP.INP**

- Dòng 1: N và M ($1 \leq N; M \leq 500$)
- Dòng thứ I trong N dòng tiếp theo, ghi M ký tự $/, \backslash$ mô tả trạng thái đoạn dây ở ô $(i; j)$ với $j = 1..M$

Kết quả: Đưa ra file văn bản **LAMP.OUT** một số duy nhất là kết quả bài toán. Nếu không có cách làm cho đèn sáng thì ghi NO SOLUTION.

Ví dụ:

LAMP.INP	LAMP.OUT
3 5 \\//\\ \\/// /\\\\\\	1

Chú ý: 40% số test tương ứng với 40% số điểm ứng với $1 \leq N \leq 4, 1 \leq M \leq 5$

BÀI 47. THANG MÁY

Để vận chuyển hòm báu vật lên phòng làm việc của mình ở tầng N trong tòa nhà chọc trời (vô số tầng), HT phải thuê công nhân bốc vác để vận chuyển hòm. Biết các chi phí liên quan đến thuê công nhân là:

- Để di chuyển theo thang bộ lên trên 1 tầng là U
- Để di chuyển theo thang bộ xuống dưới 1 tầng là D
- Để khuyên hòm vào thang máy là I
- Để khuyên hòm ra khỏi thang máy là J

Trong tòa nhà có L thang máy, việc di chuyển theo thang máy là miễn phí, vì vậy cần tận dụng chúng để giảm chi phí. Tuy nhiên cần lưu ý là mỗi thang máy chỉ dừng ở những tầng nhất định?

Yêu cầu: Hãy giúp HT tìm đường vận chuyển hòm báu vật từ tầng 1 lên tầng N với chi phí phải trả là ít nhất?

Dữ liệu: Vào từ file văn bản **LIFT.INP**

- Dòng 1: N, U, D, I, J, L
- Dòng thứ I trong số L dòng tiếp theo mô tả hoạt động của thang máy thứ I ($I = 1, 2, \dots, L$): Đầu tiên là số nguyên K_i là số lượng tầng mà thang máy thứ I sẽ dừng (các tầng được sắp xếp tăng theo thứ tự tăng dần của chỉ số).

Kết quả: Đưa ra file văn bản **LIFT.OUT** một số nguyên là chi phí nhỏ nhất tìm được?

Ví dụ:

LIFT.INP	LIFT.OUT
10 1 1 1 1 1 2 3 7	7
10 1 1 3 2 1 2 3 7	9
20 100 0 1 1 2 2 5 7 2 8 17	804

Chú ý:

- $0 \leq U, D, I, J \leq 1000, 0 \leq L \leq 500, 1 \leq N \leq 1.000.000, 2 \leq K_i \leq 1000$
- $K_1 + K_2 + \dots + K_L \leq 1000$

BÀI 48. Đảo rồng

Đất nước HP xinh đẹp là một quần đảo gồm N đảo đánh số từ 1 đến N , có M đường bay hai chiều giữa các đảo được đánh số từ 1 đến M , đường bay thứ i nối hai đảo a_i và b_i có độ dài d_i .

Việc di chuyển giữa các đảo không thực hiện bằng máy bay như bình thường mà bay bằng rồng. Mỗi đảo có một con rồng, con rồng ở đảo i có thể bay được một quãng đường xa nhất là T_i (gọi là sức bay), như vậy để con rồng ở đảo i bay được trên con đường j (nối đảo i với một đảo khác) thì $T_i \geq d_j$.

Khi di chuyển từ đảo i đến đảo j , để tiếp tục chuyển bay, bạn có thể sử dụng con rồng đang dùng hoặc đổi sang con rồng ở đảo j

Hoàng tử Dương Phong đang sống ở đảo số 1, anh ấy muốn biết hai thông tin sau để chuẩn bị cho chuyến du lịch của mình:

- Xuất phát từ đảo 1, anh ấy sử dụng con rồng tại đảo 1, không đổi rồng trong suốt hành trình thì anh ấy có thể gặp con rồng có sức bay lớn nhất là bao nhiêu?
- Xuất phát từ đảo 1, có thể đổi rồng trên đường đi thì độ dài đường đi ngắn nhất từ 1 đến N là bao nhiêu?

Dữ liệu: Vào từ file văn bản **DRAGON.INP**

- Dòng đầu tiên chứa số nguyên dương p tương ứng $p = 1$ là câu hỏi a và $p = 2$ là câu hỏi b;
- Dòng tiếp theo chứa hai số nguyên dương N, M ($1 \leq N \leq 800; 1 \leq M \leq 6.000$) theo thứ tự là số lượng đảo và số đường bay hai chiều;
- Dòng tiếp theo ghi N số nguyên dương, số thứ i là T_i ($1 \leq T_i \leq 50.000$) tương ứng là sức bay của con rồng ở đảo i ;
- M dòng tiếp theo, dòng thứ j chứa ba số nguyên u_j, v_j, d_j tương ứng là có đường bay nối hai đảo u_j, v_j và độ dài d_j ($1 \leq d_j \leq 50.000$).

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: Đưa ra file văn bản **DRAGON.OUT**

- Nếu $p = 1$ thì ghi kết quả của câu hỏi a, nếu $p = 2$ thì ghi kết quả của câu hỏi b;
- Với câu hỏi b, dữ liệu đảm bảo có thể bay từ 1 đến N và độ dài đường đi ngắn nhất là số nguyên không quá 10^9

Ví dụ:

DRAGON.INP	DRAGON.OUT	<i>Giải thích</i>
1 5 6 6 3 13 20 26 1 2 5 1 3 7 1 5 10 2 3 6 3 4 5 3 5 14	20	$p = 1$: từ đảo 1 thực hiện chuyến bay mà không đổi rồng ở các đảo thì ta có thể đi: $1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 4$. Vậy Dương Phong có thể gặp con rồng ở đảo 4 có sức bay lớn nhất là 20
DRAGON.INP	DRAGON.OUT	<i>Giải thích</i>
2 5 6 6 3 13 20 26 1 2 5 1 3 7 1 5 10 2 3 6 3 4 5 3 5 14	28	$p = 2$: từ đảo 1 thực hiện chuyến bay đến đảo N. $1 \rightarrow 2$: sử dụng rồng 1, quãng đường 5; $2 \rightarrow 3$: sử dụng rồng 1, quãng đường 6, đổi rồng; $3 \rightarrow 1$: sử dụng rồng 3, quãng đường 7; $1 \rightarrow 5$: sử dụng rồng 3, quãng đường 10. Tổng độ dài đường đi ngắn nhất là: $5 + 6 + 7 + 10 = 28$.

Chú ý: Câu hỏi a ứng với $p = 1$ được 20% số điểm của bài toán, câu hỏi b ứng với $p = 2$ được 80% số điểm của bài toán.

BÀI 49. ROADBLOCKS

Cho đồ thị vô hướng N đỉnh M cạnh.

Yêu cầu: Tìm đường đi ngắn nhì từ đỉnh 1 đến đỉnh N trên đồ thị. (đường đi ngắn nhì là đường đi có độ dài lớn hơn đường đi ngắn nhất và ngắn hơn các đường đi còn lại, một cạnh có thể sử dụng nhiều lần)

Dữ liệu: Vào từ file văn bản **ROADBLOCKS.INP**

- Dòng 1: N và M ($1 \leq N \leq 5.000$; $1 \leq M \leq 100.000$)
- M dòng tiếp theo, mỗi dòng ghi u, v, c tương ứng có cạnh nối u, v có độ dài c ($1 \leq c \leq 5.000$)

Kết quả: Đưa ra file văn bản **ROADBLOCKS.OUT** một số duy nhất là độ dài đường đi ngắn nhì

Ví dụ:

ROADBLOCKS.INP	ROADBLOCKS.OUT
4 4 1 2 100 2 4 200 2 3 250 3 4 100	450

BÀI 50. BUGS

Lịch điện tử của HT có chứa một lỗi. Vì lỗi này, số nguyên chẵn không thể được nhập vào lịch này.

HT đang lập kế hoạch cho một chuyến đi kinh doanh từ HTtown đến HTcity. Rõ ràng, HT muốn đi theo con đường ngắn nhất cho chuyến đi. Sau khi HT quay trở lại, HT sẽ phải nhập vào độ dài của đường đi vào lịch của bạn, do đó, chiều dài cần phải được một số nguyên lẻ.

Cần nhắc rằng các lỗi trong lịch này chắc chắn sẽ vẫn còn một thời gian khá dài, và rằng hệ thống đường bộ trong HTland có lẽ sẽ được xây dựng lại nhiều lần, HT đã quyết định viết một chương trình để giúp HT trong việc đối phó với vấn đề tương tự trong tương lai.

Yêu cầu: Tính chiều dài lẻ của con đường ngắn nhất từ HTtown đến HTcity hoặc kiểm tra rằng các con đường không tồn tại,

Dữ liệu: Vào từ file văn bản **BUG.INP**

- Dòng 1: 2 số nguyên dương n, m ($2 \leq n \leq 200.000$) ($0 \leq m \leq 500.000$) tương ứng là số thành phố và số con đường. Các thành phố được đánh số từ 1 đến n . HTtown là thành phố số 1 và HTcity là thành phố số n .
- M dòng tiếp theo là 3 số nguyên dương a, b, c ($1 \leq a, b \leq n, a \neq b, 1 \leq c \leq 1000$) nghĩ là đường đi từ thành phố a đến thành phố b có độ dài c .

Kết quả: Đưa ra file văn bản **BUG.OUT** một số duy nhất chiều dài lẻ của con đường ngắn nhất từ HTtown đến HTcity. HT có thể đi qua một số thành phố và 1 số con đường nhiều lần. Nếu không tồn tại đường đi, ghi số 0

Ví dụ:

BUG.INP	BUG.OUT	
6 7 1 2 1 2 6 1 1 3 1 5 6 1 3 5 2 3 4 1 5 4 4	7	

BÀI 51. Khu vui chơi

Bản đồ khu vui chơi XYZ là một hình chữ nhật có kích thước $m * n$ ô. Các hàng của hình chữ nhật được đánh số từ trên xuống dưới bắt đầu từ 1 đến m , còn các cột được đánh số từ trái qua phải, bắt đầu từ 1 đến n , ô nằm ở vị trí hàng i và cột j gọi là ô $(i; j)$. Khu vui chơi XYZ có một cổng vào đặt tại ô $(1; 1)$ và một cổng ra đặt tại ô $(m; n)$. Mỗi ô $(i; j)$ được bố trí một trò chơi, giá vé mỗi lần vào ô (I, j) là C_{ij} ($0 < C_{ij} \leq 30000$). Khách tham quan đi vào khu vui chơi từ ô $(1; 1)$. Tại mỗi ô, khách có thể di chuyển sang các ô chung cạnh với ô đó (mỗi ô có thể đi qua nhiều lần và không nhất thiết phải đi qua hết tất cả các ô), sau đó đi ra khỏi khu vui chơi từ ô $(m; n)$. Vào ngày nghỉ, nhận được bản đồ khu vui chơi XYZ với bảng giá vé tại tất cả các ô, HT đặc biệt quan tâm đến 3 trò chơi mới đặt tại 3 ô khác nhau (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , nên đã quyết định tham quan khu vui chơi XYZ và đến ít nhất hai trong 3 ô đó.

Yêu cầu: Cho bản đồ khu vui chơi XYZ với bảng giá tại tất cả các ô và vị trí 3 ô khác nhau (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , hãy tính chi phí ít nhất mà HT phải trả khi thực hiện quyết định của mình.

Dữ liệu: Vào từ file văn bản **VUICHOI.INP**

- Dòng 1: m, n ($3 \leq m, n \leq 500$)
- Dòng 2: 6 số nguyên $x_1, y_1, x_2, y_2, x_3, y_3$
- M dòng tiếp theo, mỗi dòng n số nguyên dương là giá vé tại các ô

Kết quả: Đưa ra file văn bản **VUICHOI.OUT** một số duy nhất là chi phí ít nhất mà HT phải trả

Ví dụ:

VIUCHOL.INP	VIUCHOL.OUT
3 4 1 2 1 3 1 4 1 2 9 3 7 1 1 1 3 4 5 3	13

VIUCHOL.INP	VIUCHOL.OUT
3 4 1 3 1 4 3 1 1 1 1 9 1 9 9 9 1 1 1 1	10

BÀI 52. TOURS13

Công ty du lịch X có dự án tổ chức các hành trình du lịch trong vùng lãnh thổ gồm n điểm du lịch trọng điểm, được đánh số từ 1 tới n . Hệ thống giao thông trong vùng gồm m tuyến đường một chiều khác nhau, tuyến đường thứ j ($j = 1, 2, 3, \dots, m$) cho phép đi từ địa điểm u_j đến địa điểm v_j với chi phí đi lại là một số nguyên dương $c(u_j, v_j)$. Vấn đề đặt ra cho công ty là xây dựng các hành trình du lịch cho mỗi điểm du lịch. Một hành trình du lịch cho địa điểm du lịch i phải được xây dựng sao cho xuất phát từ địa điểm i đi qua một số địa điểm khác rồi quay lại địa điểm xuất phát i với tổng chi phí (được tính như là tổng chi phí của các tuyến đường mà hành trình đi qua) nhỏ nhất.

Dữ liệu: Vào từ file văn bản **TOURS13.INP**

- Dòng đầu tiên số T là số lượng bộ dữ liệu. Tiếp đến là T nhóm dòng, mỗi dòng cho thông tin về một bộ dữ liệu theo khuôn dạng sau :
 - o Dòng thứ nhất chứa 2 số nguyên dương n, m
 - o Dòng thứ j trong số m dòng tiếp theo chứa ba số nguyên dương $u_j, v_j, c(u_j, v_j)$ cho biết thông tin về tuyến đường thứ j . Giả thiết là $u_j \neq v_j$; $c(u_j, v_j) < 10^6$; $j = 1, 2, \dots, m$

Kết quả: Đưa ra file văn bản **TOURS13.OUT** gồm T nhóm dòng tương ứng với T bộ test vào, mỗi nhóm dòng gồm n dòng, dòng thứ i ghi chi phí của hành trình du lịch cho địa điểm i . Qui ước: Ghi số -1 trên dòng i nếu không tìm được hành trình du lịch cho địa điểm i thỏa mãn yêu cầu đặt ra

Ví dụ:

TOURS13.INP	TOURS13.OUT
1	11
6 8	11
1 2 4	6
2 4 2	11
4 3 3	6
3 1 4	-1
4 1 5	
3 5 5	
5 3 1	
5 6 7	

Ràng buộc:

- Có 30% số test tương ứng với 30% số điểm của bài có $n \leq 20$.
- Có 30% số test tương ứng với 30% số điểm của bài có $20 < n \leq 100, m \leq 10^4$.
- Có 40% số test tương ứng với 30% số điểm của bài có $100 < n \leq 10^3, m \leq 10^5$

BÀI 53. PARTY2

Vương quốc HT có N thần dân sống trong N ngôi nhà đánh số từ 1 đến X. HT muốn tổ chức đại tiệc tại ngôi nhà X và mọi thần dân phải đến dự. Có M con đường 1 chiều nối 2 ngôi nhà, con đường I mất thời gian là $T[i]$. Mỗi thần dân sẽ xuất phát từ nhà mình, đi theo con đường ngắn nhất đến nhà X và trở về nhà mình sau bữa tiệc.

Với tất cả thần dân, bạn hãy cho biết thời gian lớn nhất 1 thần dân phải trả cho việc đi từ nhà đến bữa tiệc và trở về nhà.

Dữ liệu: Vào từ file văn bản **PARTY2.INP**

- Dòng 1: N, M, X ($1 \leq N \leq 1.000, 1 \leq M \leq 100.000$)
- M dòng tiếp theo, dòng thứ i mô tả con đường 1 chiều thứ $i: u, v, T[i]$ ($1 \leq T[i] \leq 100$)

Kết quả: Đưa ra file văn bản **PARTY2.OUT** một số duy nhất là đáp án bài toán?

Ví dụ:

PARTY2.INP	PARTY2.OUT	
4 8 2 1 2 4 1 3 2 1 4 7 2 1 1 2 3 5 3 1 2 3 4 4 4 2 3	10	- Thần dân 4: đi mất 3, về mất 7.

BÀI 54. THỨ 6 NGÀY 13

HSN là người mê tín, anh ta rất sợ hãi khi đối diện với “thứ 6 ngày 13”. Trong tuần, nếu ngày đi chơi với QT không phải ngày thứ 6, thì anh ta di chuyển từ nhà mình (đánh số 1) đến nhà QT (đánh số N) như bình thường. Tuy nhiên, nếu vào ngày thứ 6, thì tổng độ dài của 1 phần bất kỳ tuyến đường đi từ nhà HSN đến nhà QT phải không chia hết cho 13.

Ví dụ: HSN di chuyển từ nhà đến điểm dừng chân đầu tiên mất t_1 , từ điểm dừng đầu tiên đến điểm dừng thứ 2 mất t_2 , từ điểm thứ 2 đến nhà QT mất t_3 thì: $t_1; t_2; t_3; t_1 + t_2; t_2 + t_3; t_1 + t_2 + t_3$ phải không chia hết cho 13.

Yêu cầu: Hãy xác định đường đi ngắn nhất từ nhà HSN (đánh số 1) đến nhà QT (đánh số N)?

Dữ liệu: Vào từ file văn bản **FRIDAY13.INP**

- Dòng đầu tiên là số nguyên dương T tương ứng với số test. ($1 \leq T \leq 10$)
- Mỗi test có dạng:
 - Dòng 2: 2 số nguyên dương N và M tương ứng là N ngôi nhà và M con đường 1 chiều. ($1 \leq N \leq 50; 1 \leq M \leq 2.500$)
 - M dòng tiếp theo, mỗi dòng chứa 3 số u, v, c tương ứng có con đường 1 chiều từ nhà u đến nhà v có độ dài c . ($1 \leq c \leq 100$)
 - Dòng cuối chứa xâu ký tự “True” nếu là thứ 6, chứa xâu “False” nếu không phải thứ 6

Kết quả: Ghi ra file văn bản **FRIDAY13.OUT** trên T dòng tương ứng với T test trong file input. Dòng thứ I ghi độ dài đường đi ngắn nhất trong test i. Nếu không có đường đi thỏa mãn ghi -1

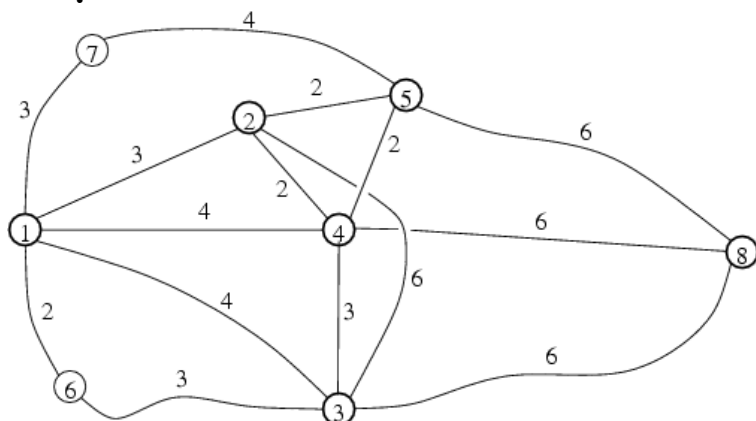
Ví dụ:

FRIDAY13.INP	FRIDAY13.OUT
3	16
5 5	-1
1 2 1	42
1 3 2	
2 4 1	
3 4 3	
4 5 11	
True	
2 2	
1 2 26	
1 2 13	
True	
3 3	
1 1 7	
1 2 26	
2 3 16	
False	

Chú ý: Có 50% số test tương ứng với dữ liệu không phải là thứ 6. Mỗi con đường có thể tham gia nhiều hơn 1 lần trong đường đi từ 1 đến N.

BÀI 55. DU LỊCH

HT land có N thành phố đánh số từ 1 đến N và M con đường 2 chiều nối chúng. HD muốn đi chuyển từ 1 đến N với yêu cầu được thăm k thành phố đánh số từ 2 đến K + 1 sao cho độ dài đường đi là ngắn nhất?

Ví dụ:

Với $k = 4$, HD muốn thăm các thành phố 2, 3, 4, 5 sao cho thành phố 2 trước 3, thành phố 4, 5 sau thành phố 3 thì đường đi ngắn nhất là con đường 1, 2, 4, 3, 4, 5, 8 với độ dài 19. Ở đây có đến thành phố 4 trước 3 nhưng thành phố này không tính vì HD có quay lại thăm 4 sau khi qua thành phố 3.

Dữ liệu: Vào từ file văn bản **DULICH.INP**

- Dòng 1: N, M, K ($2 \leq N \leq 20.000$; $1 \leq M \leq 200.000$; $0 \leq K \leq 20$)
- M dòng tiếp theo mỗi dòng ghi 3 số a, b, d tương ứng có con đường nối a và b với độ dài d ($1 \leq d \leq 1.000$)
- Dòng tiếp theo ghi số G là số lượng yêu cầu thứ tự thăm các thành phố của HD.
- G dòng tiếp theo, mỗi dòng ghi 2 số r và s tương ứng là thăm thành phố r trước khi thăm thành phố s .

Kết quả: Ghi ra file văn bản **DULICH.OUT** một số duy nhất là độ dài đường đi ngắn nhất của cuộc hành trình từ 1 đến N thỏa mãn điều kiện đầu bài?

Ví dụ:

DULICH.INP	DULICH.OUT
8 15 4	19
1 2 3	
1 3 4	
1 4 4	
1 6 2	
1 7 3	
2 3 6	
2 4 2	
2 5 2	
3 4 3	
3 6 3	
3 8 6	
4 5 2	
4 8 6	
5 7 4	
5 8 6	
3	
2 3	
3 4	
3 5	

BÀI 56. Thăm bạn

Thành Nam đang có lễ rước đức Thánh Trần nhân dịp kỷ niệm ngày giỗ của ngài. Để đảm bảo an toàn giao thông, trên các tuyến phố mà đoàn rước đi qua kể từ khi đoàn rước bắt đầu vào đầu phố cho đến khi đoàn rước đi qua hết phố, các phương tiện giao thông không được phép đi vào phố này (kể từ cả hai đầu phố). Tuy nhiên nếu có phương tiện nào đó đã ở trên phố trước khi đoàn rước đi vào phố thì nó vẫn di chuyển bình thường (kể từ cả hai đầu phố). Cũng trong khoảng thời gian đoàn rước đi trên các phố, Hùng muốn thăm một người bạn ở trong thành phố.

Thành Nam có thể được mô tả như là hệ thống giao thông gồm các tuyến phố với các điểm giao cắt là đầu mút của mỗi tuyến phố, giữa hai nút giao cắt có không quá một tuyến phố. Với mỗi tuyến phố, thời gian mà Hùng đi hết nó bằng với thời gian mà đoàn rước đi hết tuyến phố này.

Ví dụ: Nếu đoàn rước vào một tuyến phố nào đó ở thời điểm 10 và cần 5 đơn vị thời gian để đi hết tuyến phố thì Hùng chỉ có thể vào phố trước thời điểm 10 hoặc từ thời điểm 15 hay muộn hơn.

Yêu cầu: Hãy xác định khoảng thời gian ít nhất Hùng có thể đi đến đích.

Dữ liệu: Vào từ file văn bản **FESTIVAL.INP**

- Dòng đầu tiên ghi hai số nguyên dương N, M ($2 \leq N \leq 1.000, 2 \leq M \leq 10.000$) là số điểm giao cắt và số tuyến phố. Các điểm giao cắt được đánh số từ 1 đến N
- Dòng thứ hai chứa 4 số nguyên A, B, H, F với A, B vị trí xuất phát và đích đến của Hùng (Dữ liệu đảm bảo có đường đi từ A đến B), H là chênh lệch thời gian giữa thời điểm xuất phát của đoàn rước và thời điểm xuất phát của Hùng (Hùng xuất phát sau H đơn vị thời gian kể từ khi đoàn rước bắt đầu); F là số lượng điểm giao cắt có trên hành trình của đoàn rước.
- Dòng thứ ba chứa F số nguyên lần lượt là số hiệu các điểm giao cắt trên hành trình mà đoàn rước đi qua theo thứ tự. Dữ liệu đảm bảo rằng không có một tuyến phố nào mà đoàn rước đi qua nhiều hơn một lần.
- M dòng cuối cùng, mỗi dòng ghi ba số nguyên u, v và t thể hiện có một tuyến phố nối u và v với thời gian đi hết nó (của Hùng cũng như của đoàn rước) là t . Giá trị của t nằm trong khoảng $[1, 1000]$.

Kết quả: Ghi ra file văn bản **FESTIVAL.OUT** một số nguyên duy nhất là thời gian ngắn nhất mà Hùng có thể đi từ A đến B .

Ví dụ:

FESTIVAL.INP	FESTIVAL.OUT
6 5 1 6 20 4 5 3 2 4 1 2 2 2 3 8 2 4 3 3 6 10 3 5 15	21

BÀI 57. CẢNH SÁT

Để truy bắt tội phạm, cảnh sát xây dựng một hệ thống máy tính mới. Bản đồ khu vực bao gồm N thành phố và E đường nối 2 chiều. Các thành phố được đánh số từ 1 đến N . Cảnh sát muốn bắt các tội phạm di chuyển từ thành phố này đến thành phố khác. Các điều tra viên, theo dõi bản đồ, phải xác định vị trí thiết lập trạm gác. Hệ thống máy tính mới phải trả lời được 2 loại truy vấn sau:

1. Đối với hai thành phố A, B và một đường nối giữa hai thành phố G_1, G_2 , hỏi tội phạm có thể di chuyển từ A đến B nếu đường nối này bị chặn (nghĩa là tên tội phạm không thể sử dụng con đường này) không?

2. Đối với 3 thành phố A, B, C , hỏi tội phạm có thể di chuyển từ A đến B nếu như toàn bộ thành phố C bị kiểm soát (nghĩa là tên tội phạm không thể đi vào thành phố này) không?

Dữ liệu: Vào từ file văn bản **POLICE.INP**

- Dòng đầu tiên chứa 2 số nguyên N và E ($2 \leq N \leq 100000, 1 \leq E \leq 500000$), số thành phố và số đường nối.
- Mỗi dòng trong số E dòng tiếp theo chứa 2 số nguyên phân biệt thuộc phạm vi $[1, N]$ - cho biết nhãn của hai thành phố nối với nhau bởi một con đường. Giữa hai thành phố có nhiều nhất một đường nối.
- Dòng tiếp theo chứa số nguyên Q ($1 \leq Q \leq 300\,000$), số truy vấn được thử nghiệm trên hệ thống.
- Mỗi dòng trong Q dòng tiếp theo chứa 4 hoặc 5 số nguyên. Số đầu tiên cho biết loại truy vấn - 1 hoặc 2.
 - o Nếu loại truy vấn là 1, tiếp theo trên cùng dòng là 4 số nguyên A, B, G_1, G_2 với ý nghĩa như đã mô tả. A khác B ; G_1, G_2 mô tả một con đường có sẵn.
 - o Nếu loại truy vấn là 2, tiếp theo trên cùng dòng là 3 số nguyên A, B, C với ý nghĩa như đã mô tả. A, B, C đôi một khác nhau.

Dữ liệu được cho sao cho ban đầu luôn có cách di chuyển giữa hai thành phố bất kỳ.

Kết quả: Ghi ra file văn bản **File POLICE.OUT** gồm Q dòng, mỗi dòng chứa câu trả lời cho một truy vấn. Nếu câu trả lời là khẳng định, in ra **"yes"**. Nếu câu trả lời là phủ định, in ra **"no"**.

Ví dụ:

POLICE.INP	POLICE.OUT
13 15	yes
1 2	yes
2 3	yes
3 5	no
2 4	yes
4 6	
2 6	
1 4	
1 7	
7 8	
7 9	
7 10	
8 11	
8 12	
9 12	
12 13	
5	
1 5 13 1 2	
1 6 2 1 4	
1 13 6 7 8	
2 13 6 7	
2 13 6 8	

BÀI 57. WEATHER

Hãng hàng không OlympAirways thực hiện các chuyến bay giữa N sân bay được đánh số từ 1 đến N. Hệ thống các chuyến bay của hãng hàng không được thiết lập sao cho từ một sân bay bất kỳ được phục vụ bởi hãng luôn có thể bay đến bất kì sân bay còn lại bằng cách sử dụng chuyến bay trực tiếp giữa 2 sân bay hoặc sử dụng nhiều hơn một chuyến bay. Mỗi chuyến bay thực hiện việc di chuyển giữa 2 thành phố theo cả 2 chiều.

Có một vấn đề nảy sinh hoàn toàn từ yếu tố khách quan là: Một chuyến bay không thể thực hiện được trong những khoảng thời gian nhất định do thời tiết xấu. Vì thế, rất có thể hành khách không thể bay từ sân bay A đến sân bay B mà chỉ sử dụng các chuyến bay của hãng. Để khảo sát các tình huống như vậy, Trung tâm CNTT của hãng đưa ra khái niệm độ dính kết giữa cặp 2 sân bay A và B được xác định như là số lượng các

chuyến bay mà việc không thực hiện một trong số chúng (còn các chuyến bay khác vẫn thực hiện bình thường) dẫn đến không thể bay từ sân bay A đến sân bay B.

Yêu cầu: Hãy tính tổng tất cả các độ dính kết giữa mọi cặp sân bay

Dữ liệu: Vào từ file văn bản **WEATHER.INP**

- Dòng đầu tiên chứa số nguyên N ($1 \leq N \leq 100$)
- Dòng thứ 2 chứa số nguyên M ($1 \leq M \leq 5000$) – số lượng các chuyến bay
- Mỗi một trong số M dòng tiếp theo chứa thông tin về một chuyến bay bao gồm hai số nguyên dương trong khoảng từ 1 đến N chỉ số của hai sân bay được nối bởi chuyến bay.

Kết quả: Ghi ra file văn bản **WEATHER.OUT** Ghi 1 số nguyên dương duy nhất là tổng độ dính kết giữa mọi cặp sân bay (A, B) mà trong đó $A \leq B$.

Ví dụ:

WEATHER.INP	WEATHER.OUT
11	102
13	
2 10	
10 6	
8 11	
2 4	
11 10	
5 4	
6 11	
8 2	
9 4	
11 2	
1 7	
7 4	
1 3	

BÀI 58. LIÊN THÔNG

Cho một đồ thị vô hướng gồm n đỉnh đánh số từ 1 tới n và m cạnh đánh số từ 1 tới m . Cạnh thứ i nối giữa hai đỉnh u_i, v_i . Nếu ta xoá đi một đỉnh nào đó của đồ thị, số thành phần liên thông của đồ thị có thể tăng lên. Nhiệm vụ của bạn là với mỗi đỉnh, hãy tính xem nếu ta xoá đỉnh đó đi thì đồ thị mới nhận được có bao nhiêu thành phần liên thông.

Dữ liệu: Vào từ file văn bản **GRAPH.INP**

- Dòng đầu chứa hai số nguyên dương n, m ($n \leq 20000; m \leq 50000$)
- m dòng sau, dòng thứ i chứa hai số nguyên dương u_i, v_i .

Kết quả: Ghi ra file văn bản **GRAPH.OUT** n dòng, dòng thứ j cho biết số thành phần liên thông của đồ thị nếu ta xóa đi đỉnh j .

Ví dụ

GRAPH.INP	GRAPH.OUT
4 3	1
1 2	3
2 3	1
2 4	1

Chú ý: Ít nhất 60% số điểm ứng với các test có $n \leq 1000$; $m \leq 2000$

BÀI 59. MA CÀ RỒNG

Một con ma cà rồng muốn di chuyển từ địa điểm A đến địa điểm B. Nó có thể di chuyển giữa hai địa điểm bằng đường đi trên mặt đất (và phải chịu đựng ánh nắng mặt trời) hoặc tránh nắng bằng việc sử dụng các đường hầm. Bạn có bản đồ mô tả các đường đi trên mặt đất và hầm giữa các địa điểm trong khu vực. Cả đường đi trên mặt đất và đường hầm đều có thể đi theo hai chiều. Khó khăn ở chỗ ma cà rồng chỉ có thể đi dưới ánh nắng tổng cộng không quá s giây.

Yêu cầu: Hãy tính thời gian ngắn nhất để Ma cà rồng có thể đi từ A đến B.

Dữ liệu: Vào từ file văn bản **VAMPIRE.INP**

- Dòng 1: chứa số nguyên s ($0 \leq s \leq 3600$).
- Dòng 2: Ghi 2 số nguyên dương n và m là số điểm chốt và tổng số đường và hầm nối giữa các điểm chốt ($1 \leq n \leq 1600$, $1 \leq m \leq 10000$).
- m dòng tiếp: Mỗi dòng mô tả một đường hầm hoặc một đường đi trên mặt đất với 4 số nguyên u, v, t, ok : ok bằng 1 nếu là đường đi trên mặt đất và 0 nếu là đường hầm; u và v là 2 địa điểm được nối với nhau, t là thời gian đi hết đường. ($0 \leq u, v \leq n - 1$; $1 \leq t \leq 10000$) các địa điểm được đánh số từ 0 đến $n - 1$).

Kết quả: Ghi ra file văn bản **VAMPIRE.OUT** Ghi -1 nếu bạn không thể đi từ 0 đến $n - 1$, ngược lại ghi ra thời gian nhỏ nhất bạn tìm được

Ví dụ

VAMPIRE.INP	VAMPIRE.OUT
3	9
4 6	
0 1 3 1	
0 2 4 1	
0 3 10 1	
1 2 3 0	
1 3 1 1	
2 3 3 0	

Giải thích: Đường đi tối ưu là $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ với tổng thời gian là 9 giây và 3 giây dưới ánh nắng.

Chú ý:

- Ít nhất 30% số điểm ứng với các test có $n \leq 50$;
- Ít nhất 50% số điểm ứng với các test có $n \leq 200$.

BÀI 60. HD nổi loạn

Có N thị trấn ở HT land. Một vài thị trấn được nối với nhau bằng những con đường trực tiếp 2 chiều. Những con đường này không cắt nhau ở bên ngoài các thị trấn này. Mỗi cặp thị trấn có thể được nối bằng nhiều nhất một con đường trực tiếp. Bạn có thể xuất phát từ 1 thị trấn này đến một thị trấn khác một cách trực tiếp thông qua 1 con đường hoặc gián tiếp theo 1 đường đi.

Trên HT land, có một nhóm HD phản loạn, nhóm này sẽ ngăn chặn tất các tuyến đường ra vào một thị trấn nào đó, khi họ làm như vậy, sẽ có một số cặp thị trấn không thể di chuyển đến nhau.

Bạn được HT nhờ xác định xem có bao nhiêu cặp thị trấn không thể di chuyển đến nhau được khi HD làm loạn ở lần lượt các thị trấn?

Dữ liệu: Vào từ file văn bản **BLO.INP**

- Dòng 1: N và M là số thị trấn và số tuyến đường hai chiều nối giữa hai thị trấn ($1 \leq N \leq 100.000, 1 \leq M \leq 500.000$)
- M dòng tiếp theo, mỗi dòng ghi 2 số a và b thể hiện một con đường 2 chiều nối giữa thị trấn a và b . ($1 \leq a, b \leq N; a \neq b$)

Kết quả: Ghi ra file văn bản **BLO.OUT** trên N dòng, dòng thứ i ghi số cặp thị trấn mà không thể di chuyển đến nhau nếu cuộc nổi loạn diễn ra ở thành phố i .

Ví dụ:

BLO.INP	BLO.OUT
5 5	8
1 2	8
2 3	16
1 3	14
3 4	8
4 5	

BÀI 61. JOB

HT đang hết tiền và cần đi làm gấp để kiếm thu nhập. Vương quốc HT có C thành phố (đánh chỉ số từ 1 đến C) và P con đường 1 chiều nối chúng, con đường thứ I nối thành phố A_i với thành phố B_i với chi phí di chuyển là 0 VND, tại mỗi thành phố, HT có thể kiếm được tối đa D VND.

Để giúp HT kiếm được nhiều tiền hơn, hãng hàng không HD xây dựng F chuyến bay 1 chiều, mỗi chuyến bay nối thành phố J_i với thành phố K_i với chi phí A_i với thành phố

T_i . HT có thể sử dụng các chuyến bay này và trả tiền sau cùng (trả 1 cục sau khi kiểm được nhiều tiền)

Hiện tại HT đang ở thành phố S và bắt đầu công việc kiếm tiền của mình, HT có thể kết thúc công việc của mình ở bất kỳ thành phố nào mà HT muốn.

Bạn hãy giúp HT tính xem anh ta có thể kiếm được tối đa bao nhiêu tiền?

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

- Dòng 1: D, P, C, F, S ($1 \leq D \leq 1.000, 1 \leq P \leq 150, 2 \leq C \leq 220, 1 \leq F \leq 350$)
- P dòng tiếp theo, dòng thứ I ghi 1 số a và b tương ứng có con đường 1 chiều nối thành phố a với thành phố b
- F dòng tiếp theo, dòng thứ I ghi 3 số a, b và t tương ứng có chuyến bay 1 chiều từ thành phố a đến thành phố b với chi phí t ($1 \leq t \leq 50.000$)

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: ghi ra file văn bản **JOB.OUT** một số nguyên duy nhất là số tiền nhiều nhất HT kiếm được, in ra -1 nếu HT có thể kiếm được vô hạn tiền.

Ví dụ:

JOB.INP	JOB.OUT	
100 3 5 2 1 1 5 2 3 1 4 5 2 150 2 5 120	250	1→5→2→3

BÀI 62. HOLES

Vương quốc HT có N thành phố và M con đường 2 chiều nối giữa 2 thành phố. Tại một số thành phố có 1 “lỗ hồng thời gian”, khi bạn bước vào lỗ hồng này ở thành phố u, bạn sẽ di chuyển 1 chiều đến thành phố v ở thời điểm trước khi bạn bước vào thành phố u (quay ngược thời gian). Tổng số có W “lỗ hồng thời gian”

HD là một người ưa mạo hiểm, anh ta muốn mình xuất phát từ 1 thành phố nào đó, đi qua một số con đường 2 chiều và một số “lỗ hồng thời gian” sau đó trở về thành phố xuất phát vào trước thời điểm anh ta bắt đầu bước vào cuộc hành trình (gặp lại chính mình ở thời điểm xuất phát).

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

- Dòng 1: Số test F ($1 \leq F \leq 5$)
 - o Dòng tiếp theo: N, M, W ($1 \leq N \leq 500; 1 \leq M \leq 2.500; 1 \leq W \leq 200$)
 - o M dòng tiếp theo, mỗi dòng ghi 3 số u, v, t tương ứng có con đường 2 chiều bình thường nối thành phố u với thành phố v mất thời gian t (s) ($t \leq 10.000$). 2 thành phố có thể có nhiều hơn 1 con đường nối chúng.
 - o W dòng tiếp theo, mỗi dòng ghi 3 số u, v, t tương ứng có “lỗ hổng thời gian” di chuyển từ thành phố u với thành phố v trở lại quá khứ trước t (s) ($t \leq 10.000$)

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: ghi ra file văn bản **HOLES.OUT** tương ứng với mỗi test, ghi YES nếu có thể thực hiện được cuộc hành trình, ghi NO nếu không

Ví dụ:

HOLES.INP	HOLES.OUT	
2	NO	Test 1: Không thực hiện được.
3 3 1	YES	Test 2: $1 \rightarrow 2 \rightarrow 3 \rightarrow 1$
1 2 2		
1 3 4		
2 3 1		
3 1 3		
3 2 1		
1 2 3		
2 3 4		
3 1 8		

BÀI 62. Shopping

Trong thành phố X, Có N điểm bán hàng và M con đường hai chiều nối chúng.

Có K loại cá khác nhau được bán ở thành phố X. HD và HP muốn mua tất cả các loại cá này bằng cách như sau:

- HD và HP chọn cho bản thân mình một con đường đi từ điểm bán hàng số 1 đến điểm bán hàng số N . Chú ý rằng đường đi của HD không nhất thiết phải giống đường đi của HP

- Trên đường đi, cả hai sẽ mua các loại cá trên đường đi của mình
- Khi kết thúc ở điểm bán hàng N, cả hai sẽ gộp chung cá của mình lại cho đủ K loại cá
- Cả hai có thể đến thành phố N ở giữa quá trình đi mua cá nhưng cả hai phải kết thúc hành trình ở điểm bán hàng N
- Nếu một trong hai người đến điểm N trước thì sẽ ngồi đợi người còn lại đến. Vậy thời gian mua hàng bằng MAX trong 2 thời gian của hai người

Yêu cầu: Tìm thời gian nhỏ nhất để cả hai người mua đủ K loại cá?

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

- Dòng 1: N, M, K ($2 \leq N \leq 10^3, 1 \leq M \leq 2.10^3, 1 \leq K \leq 10$)
- N dòng tiếp theo, dòng i mô tả loại cá bán ở điểm i gồm:
 - o $T_i = j$: Số loại cá bán tại điểm i
 - o T_i số nguyên A_{ij} tiếp theo mô tả loại cá bán tại điểm i
- M dòng tiếp theo mỗi dòng chứa 3 số nguyên u, v, t tương ứng có cạnh u, v thời gian di chuyển t .
- Đồ thị luôn liên thông. Không có 2 đỉnh nối bởi 2 cạnh, không có cạnh tự nối 1 đỉnh với chính nó.

Kết quả: ghi ra file văn bản **SHOPPING.OUT** một số nguyên duy nhất là thời gian ngắn nhất.

Ví dụ

SHOPPING.INP	SHOPPING.OUT	
5 5 5 1 1 1 2 1 3 1 4 1 5 1 2 10 1 3 10 2 4 10 3 5 10 4 5 10	30	1 → 2 → 4 → 5 1 → 3 → 5

BÀI 63. PATH1

Cho bảng $n \times m$, ô trái trên $(0,0)$, ô phải dưới là ô $(n-1, m-1)$. Từ một ô có thể di chuyển sang ô kề cạnh.

Yêu cầu: Cho q câu hỏi dạng x, y, u, v . Tìm đường đi ngắn nhất từ $(x, y) \rightarrow (u, v)$

Dữ liệu: vào từ file văn bản **PATH1.INP**

- Dòng 1: n, m ($1 \leq n \leq 7, 1 \leq m \leq 5 \times 10^3$)
- n dòng tiếp theo, mỗi dòng m số nguyên mô tả bảng đã cho ($0 \leq a_{ij} \leq 3 \times 10^3$)

- Dòng tiếp theo q ($1 \leq q \leq 3 \times 10^4$)
- q dòng tiếp theo mỗi dòng 4 số nguyên x, y, u, v

Kết quả: ghi ra file văn bản **PATH1.OUT** ghi trên q dòng, dòng thứ i là đáp án câu hỏi i

Ví dụ:

PATH1.INP	PATH1.OUT
3 5	1
0 0 0 0 0	1
1 9 9 9 1	18
0 0 0 0 0	
3	
0 0 2 4	
0 3 2 3	
1 1 1 3	

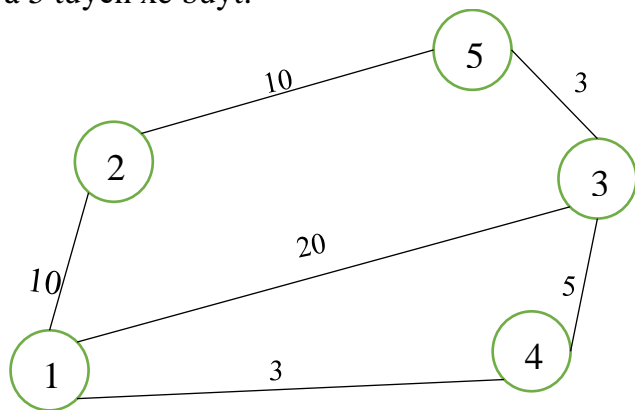
BÀI 64. Vé xe miễn phí

An là học sinh sống ở thành phố Hà Nội, hàng ngày em phải đi từ nhà tới trường bằng xe buýt. Thành phố có n nút giao thông được đánh số từ 1 đến n và m tuyến xe buýt hai chiều. Mỗi nút giao thông i, j có không quá một tuyến xe buýt hai chiều, nếu có thì để đi từ nút i đến nút j (hoặc từ nút j đến nút i) với giá vé là $t_{ij} = t_{ji}$ đồng. Nhà của An nằm ở nút giao thông 1 còn trường học lại ở nút giao thông n . Để lựa chọn đường đi từ nhà đến trường An luôn chọn theo đường đi với chi phí ít nhất.

Ví dụ: thành phố có 5 nút giao thông và 5 tuyến xe buýt:

- Tuyến 1: 1-2 giá vé 10 đồng;
- Tuyến 2: 2-5 giá vé 10 đồng;
- Tuyến 3: 1-4 giá vé 3 đồng;
- Tuyến 4: 3-4 giá vé 5 đồng;
- Tuyến 5: 3-5 giá vé 3 đồng;
- Tuyến 6: 1-3 giá vé 20 đồng;

Đường đi $1 \rightarrow 4 \rightarrow 3 \rightarrow 5$ hết 11 đồng là ít nhất.



Nhân kỷ niệm 110 năm thành lập trường An có nhận được một vé đi xe buýt miễn phí. Vé có thể dùng để đi xe buýt miễn phí một lần trên một tuyến bất kỳ. Với vé xe miễn phí này An muốn biết chi phí ít nhất để đi từ nhà đến trường là bao nhiêu? Với ví dụ trên, đi $1 \rightarrow 3 \rightarrow 5$ có sử dụng vé xe miễn phí (tuyến 1-3) hết 3 đồng là ít nhất;

Yêu cầu: Cho biết các tuyến xe buýt và giá vé tương ứng. Tìm chi phí ít nhất để từ nhà (nút giao thông 1) đến trường học (nút giao thông n) với vé xe miễn phí mà An có.

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

- Dòng đầu tiên ghi hai số nguyên dương n và m ($3 \leq n \leq 5000, m \leq 30000$)
- m dòng sau, mỗi dòng gồm 3 số nguyên i, j, t_{ij} ($1 \leq i, j \leq n, 0 < t_{ij} \leq 30000$) mô tả có tuyến xe buýt $i - j$ đi hết t_{ij} đồng.

- Dữ liệu đảm bảo có đường đi từ 1 đến n

Kết quả: ghi ra file văn bản **VEXE.OUT** chi phí ít nhất để đi từ nhà đến trường với vé xe miễn phí mà An có.

Ví dụ:

VEXE.INP	VEXE.OUT
5 6	3
1 2 10	
2 5 10	
1 4 3	
3 4 5	
3 5 3	
1 3 20	

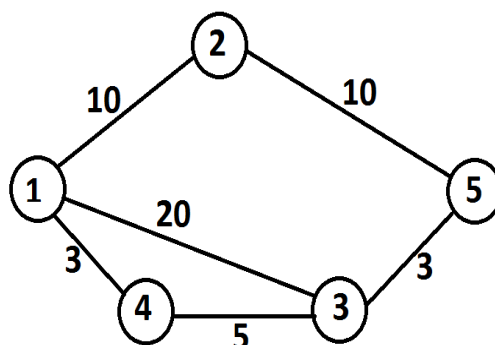
VEXE.INP	VEXE.OUT
5 5	6
1 2 10	
2 5 10	
1 4 3	
3 4 5	
3 5 3	

Chú ý: 50% số test, tương ứng với 50% số điểm có $n \leq 100$

BÀI 65. Vé xe miễn phí

Tham gia trò chơi nhảy lò cò, thật may mắn, Khuê đã giành giải nhất của cuộc thi. Phần thưởng mà Khuê nhận được là k vé xe buýt miễn phí để đi thăm quan thành phố Hạ Long. Mỗi vé xe chỉ được sử dụng một lần và có thể sử dụng cho bất kỳ tuyến xe buýt nào trong thành phố. Thành phố có n nút giao thông được đánh số từ 1 đến n và m tuyến xe buýt hai chiều. Mỗi cặp nút giao thông i, j có không quá một tuyến xe buýt hai chiều, nếu có thì để đi từ nút i đến nút j (hoặc từ nút j đến nút i) với giá vé là $c_{ij} = c_{ji}$ đồng. Xuất phát từ nút giao thông s , Khuê muốn di chuyển đến nút giao thông t và anh luôn lựa chọn đường đi với chi phí ít nhất.

Ví dụ: thành phố có 5 nút giao thông và 6 tuyến xe buýt:



Tuyến 1: 1-2 giá vé 10 đồng;

Tuyến 2: 2-5 giá vé 10 đồng;

Tuyến 3: 1-4 giá vé 3 đồng;

Tuyến 4: 3-4 giá vé 5 đồng;

Tuyến 5: 3-5 giá vé 3 đồng;

Tuyến 6: 1-3 giá vé 20 đồng.

Xuất phát từ nút 1 đến nút 5, đi theo hành trình $1 \rightarrow 4 \rightarrow 3 \rightarrow 5$ hết 11 đồng là đường đi với chi phí ít nhất. Tuy nhiên, nếu Khuê sử dụng 1 vé xe miễn phí thì đường đi $1 \rightarrow 3 \rightarrow 5$ hết 3 đồng là ít nhất (vé xe miễn phí được sử dụng tại tuyến 1-3).

Yêu cầu: Cho biết các tuyến xe buýt với giá vé tương ứng và các giá trị s, t, k . Hãy tính chi phí ít nhất để đi từ nút giao thông s đến nút giao thông t mà không sử dụng quá k vé xe miễn phí.

Dữ liệu: Vào từ file văn bản **FREEBUS.INP**:

- Dòng đầu tiên ghi năm số nguyên dương n, m, k, s, t ;
- m dòng sau, mỗi dòng 3 số nguyên i, j, c_{ij} mô tả có tuyến xe buýt $i - j$ hết c_{ij} đồng.

Kết quả: Đưa ra file văn bản **FREEBUS.OUT** một số duy nhất là chi phí ít nhất để đi từ nút giao thông s đến nút giao thông t mà không sử dụng quá k vé xe miễn phí.

Ví dụ:

FREEBUS.INP	FREEBUS.OUT
5 6 1 1 5	3
1 2 10	
2 5 10	
1 4 3	
3 4 5	
3 5 3	
1 3 20	

Ghi chú:

- Có 40% số test ứng với 40% số điểm có $n \leq 100, m \leq 1000$ và $k = 1$;
- Có 20% số test ứng với 20% số điểm có $n \leq 10^5, m \leq 10^5$ và $k = 1$;
- Có 40% số test còn lại ứng với 40% số điểm có $n \leq 10^5, m \leq 10^5$ và $k \leq 5$.

BÀI 66. Mario cứu công chúa

Mario giải cứu công chúa bị nhốt trong một kho chứa hàng tại bến cảng; Trong kho chứa các kiện hàng hình chữ nhật; Bên trên các kiện hàng luôn tồn tại nguy cơ tấn công nên chàng không thể nhảy lên và đi ở phía trên; Bản đồ bến cảng được mô tả trong góc phần tư thứ nhất của mặt phẳng tọa độ OXY. Các kiện hàng được xếp sao cho các hình chữ nhật đáy có các cạnh song song với các trục tọa độ và giữa hai kiện hàng bất kỳ luôn có lối đi. Các hình chữ nhật đáy của các kiện hàng được mô tả bởi tọa độ đỉnh trái trên và đỉnh phải dưới. Mario muốn di chuyển từ điểm có tọa độ $(x_1; y_1)$ đến điểm có tọa độ $(x_2; y_2)$, để nhanh chóng cứu công chúa Mario phải tìm một đường đi nào đó ngắn nhất mà không được trèo qua các kiện hàng đang đặt trong kho.

Yêu cầu: Hãy giúp anh chàng Mario tìm đường đi ngắn nhất từ điểm $(x_1; y_1)$ đến điểm $(x_2; y_2)$ trong kho.

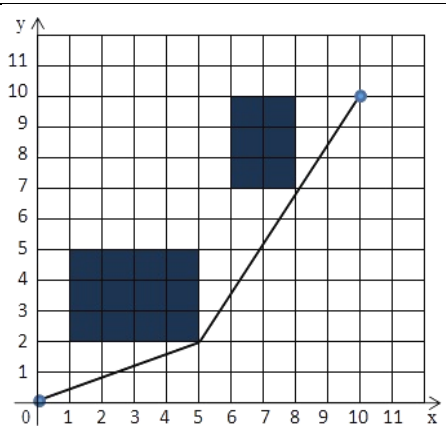
Dữ liệu: vào từ file văn bản **MARIO.INP**

- Dòng đầu là số n ($1 \leq n \leq 200$);
- Dòng thứ hai là x_1, y_1 ;
- Dòng thứ ba là x_2, y_2 ;
- Dòng thứ i trong n dòng tiếp theo ghi bốn số lx_i, ly_i, rx_i, ry_i ($lx_i < rx_i, ly_i > ry_i$) trong đó $(lx_i; ly_i)$ là tọa độ góc trái trên, $(rx_i; ry_i)$ là tọa độ góc phải dưới

của hình chữ nhật đáy của công-tơ-nơ thứ i . Các tọa độ có giá trị nguyên không âm, không vượt quá 10^4 .

Kết quả: ghi ra file văn bản **MARIO.OUT** một số duy nhất là độ dài đường đi ngắn nhất từ điểm $(x_1; y_1)$ đến điểm $(x_2; y_2)$. Kết quả lấy độ chính xác 3 chữ số sau dấu chấm thập phân.

Ví dụ:

MARIO.INP	MARIO.OUT	Giải thích
2 0 0 10 10 1 5 5 2 6 10 8 7	14.819	

BÀI 67. Cẩm đường

Hệ thống giao thông thành phố Định Nam gồm n nút giao thông đánh số $1, 2, \dots, n$ kết nối bởi m đường hai chiều, mỗi con đường tốn thời gian di chuyển nhất định, lượng thời gian này đúng với mọi phương tiện giao thông vì thành phố quy định tốc độ di chuyển cố định trong toàn hệ thống.

Trong kỳ thi chọn HSG DHBB, Nam cần di chuyển từ nút giao thông s đến nút giao thông t . Tuy nhiên, khi chuẩn bị xuất phát thì cậu mới biết do có đoàn ngoại giao nước ngoài đến làm việc nên một số con đường sẽ bị chặn tạm thời. Cụ thể, hành trình của đoàn ngoại giao là (w_1, w_2, \dots, w_3) hành trình này đảm bảo thông suốt và không có con đường nào xuất hiện quá một lần.

Đối với mỗi con đường trong hành trình, tạm kí hiệu là (a, b) , kể từ khi đoàn ngoại giao xuất hiện ở a cho đến khi ra khỏi b , không có phương tiện nào được phép tiến vào cả từ a lẫn b , các phương tiện có mặt trên đường từ trước đó vẫn có thể tham gia giao thông như bình thường. Nam xuất phát muộn hơn đoàn ngoại giao k phút, hãy xác định thời điểm cậu đến đích sớm nhất có thể.

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

- Dòng 1: hai số nguyên n, m ($2 \leq n \leq 10^3$; $2 \leq m \leq 10^4$).
- Dòng 2: bốn số nguyên s, k, c ($0 \leq k, c \leq 10^3$).
- Dòng 3: c số nguyên w_1, w_2, \dots, w_3 .
- Dòng 4 ... $m + 3$: mỗi dòng ba số nguyên u, v, l ($1 \leq l \leq 10^3$) thể hiện một con đường nối hai nút giao thông u, v tốn thời gian di chuyển l phút, không có cặp nút giao thông nào được nối bởi nhiều hơn một con đường.

Kết quả: ghi ra file văn bản **DISENTR.OUT** số phút di chuyển tối thiểu của Bòm.

Ví dụ

DISENTR.INP	DISENTR.OUT
6 5 1 6 20 4 5 3 2 4 1 2 2 2 3 8 2 4 3 3 6 10 3 5 15	21
8 9 1 5 5 5 1 2 3 4 5 1 2 8 2 7 4 2 3 10 6 7 40 3 6 5 6 8 3 4 8 4 4 5 5 3 4 23	40

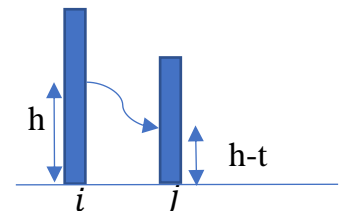
Chú ý:

- 30% số điểm của bài tương ứng với $1 \leq m \leq 1000$;
- 30% số điểm của bài tương ứng với $1000 < m \leq 5000$;
- 40% số điểm còn lại không có ràng buộc gì thêm.

BÀI 68. ROBOT

HD vừa sáng tạo ra một trò chơi điều khiển robot mới cho 2 bé Bi, Bo chơi. Nội dung trò chơi như sau:

- Có N cây cột đánh số từ 1 đến N , cây cột thứ i có chiều cao $h[i](m)$
- Có M đường nhảy dạng i, j, t tương ứng là nhảy từ cây i sang cây j (hoặc từ cây j sang cây i) mất $t(s)$ và nếu nhảy từ độ cao h ($h \in N, h \leq h[i]$) của cây i thì sang cây j sẽ có độ cao là $h - t$ với điều kiện $0 \leq h - t \leq h[j]$
- Nếu robot di chuyển lên xuống trên cột hiện tại, thời gian di chuyển mất $1(s)$ trên $1m$ di chuyển.



Hiện tại robot đang ở độ cao X của cây 1, Bi-Bo cần phải tìm phương án di chuyển nhanh nhất đến độ cao $h[N]$ của cây N . Bạn hãy giúp 2 bé Bi-Bo tính thời gian di chuyển ngắn nhất thỏa mãn yêu cầu đầu bài?

Dữ liệu: Vào từ file văn bản **ROBOT.INP**

- Dòng 1: Chứa 3 số nguyên dương N, M, X tương ứng là số lượng cây cột, số lượng đường nhảy và độ cao của robot đang ở cột 1. ($2 \leq N \leq 100.000; 1 \leq M \leq 300.000; 0 \leq X \leq h[1]$)
- N dòng tiếp theo, dòng thứ i chứa 1 số nguyên dương $h[i]$ tương ứng là chiều cao của cột i . ($1 \leq h[i] \leq 1.000.000.000 \forall i = 1..N$)
- M dòng tiếp theo, mỗi dòng chứa 3 số nguyên dương i, j, t tương ứng là nhảy từ cây i sang cây j (hoặc từ cây j sang cây i) mất $t(s)$ ($1 \leq t \leq 1.000.000.000$)

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản **ROBOT.OUT** một số duy nhất là thời gian ngắn nhất để robot di chuyển đến độ cao $h[N]$ của cây N , nếu không thể di chuyển đến thì ghi -1

Ví dụ:

ROBOT.INP	ROBOT.OUT	Giải thích
5 5 0 50 100 25 30 10 1 2 10 2 5 50 2 4 20 4 3 1 5 4 20	110	Trèo lên 50(m) ở cây 1 mất 50(s) Nhảy từ cây 1 sang cây 2: - Mất 10(s) - ở độ cao 40 trên cây 2 Nhảy từ cây 2 sang cây 4: - Mất 20(s) - ở độ cao 20 trên cây 4 Nhảy từ cây 4 sang cây 5: - Mất 20(s) - ở độ cao 0 - trèo thêm 10(m) mất 10(s) Tổng thời gian: 110(s)

ROBOT.INP	ROBOT.OUT	Giải thích
2 1 0 1 1 1 2 100	-1	Từ cây 1, bất kỳ độ cao nào, khi nhảy sang cây 2 đều không thực hiện được vì $h - t < 0$

ROBOT.INP	ROBOT.OUT	Giải thích
4 3 30 50 10 20 50 1 2 10 2 3 10 3 4 10	100	Di chuyển xuống 10(m) ở cây 1 mất 10 (s) và đang ở độ cao 20(m) Nhảy sang cây 2: - Mất 10(s) - ở độ cao 10 trên cây 2. Nhảy sang cây 3: - Mất 10(s) - Ở độ cao 0(m), trèo lên 10(m) mất 10(s), ở độ cao 10(m); Nhảy sang cây 4: - Mất 10(s), - ở độ cao 0 (m), trèo lên 50(m) mất 50(s) Tổng thời gian: 100(s)

Chú ý:

- 25% số điểm tương ứng với các test có: $N \leq 1.000$; $M \leq 3.000$; $h[i] \leq 100 \forall i = 1..N$; $t \leq 100$
- 25% số điểm tương ứng với các test có $X = 0$

BÀI 69. Xây dựng thư viện

Có N thành phố và M con đường hai chiều nối chúng, đã bị hỏng, cần sửa chữa. Chính phủ muốn:

- Mỗi thành phố có 1 thư viện, hoặc:
- Có thể di chuyển đến thành phố có thư viện.

Biết rằng ban đầu đất nước chưa có 1 thư viện nào, các con đường đều phải sửa chữa, chi phí xây thư viện là c_{lib} , chi phí sửa đường là c_{road}

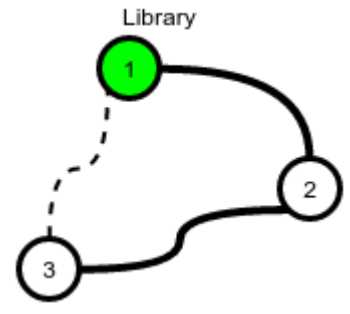
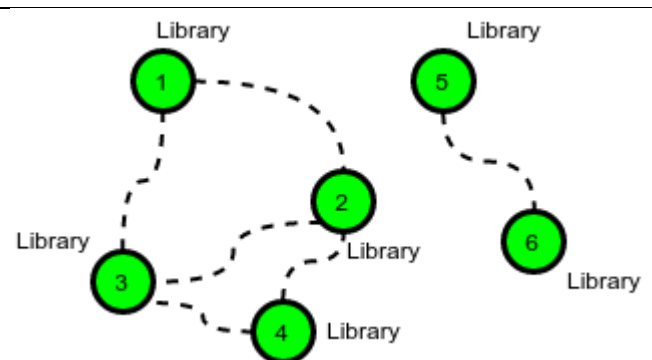
Hãy tìm các xây dựng sao cho chi phí nhỏ nhất?

Dữ liệu: Vào từ file văn bản **LIB.INP**

- Dòng 1: Q ($1 \leq Q \leq 10$) trong đó Q là số test
- Mỗi test được mô tả:
 - o Dòng 1: n, m, c_{lib}, c_{road} ($1 \leq n \leq 10^5, 0 \leq m \leq \min(10^5, \frac{n(n-1)}{2}), 1 \leq c_{lib}, c_{road} \leq 10^5$)
 - o m dòng tiếp theo, mỗi dòng 2 số nguyên dương u, v ($1 \leq u, v \leq n$) tương ứng có cạnh nối thành phố u và v

Kết quả: Ghi ra file văn bản **LIB.OUT** trên Q dòng, dòng i là đáp án test i

Ví dụ:

LIB.INP	LIB.OUT	
2 3 3 2 1 1 2 3 1 2 3 6 6 2 5 1 3 3 4 2 4 1 2 2 3 5 6	4 12	 

BÀI 70. PATH

Cho đồ thị vô hướng N đỉnh và M cạnh, mỗi cạnh M_i thì có trọng số C_i

Độ dài đường đi trên các cạnh M_1, M_2, \dots, M_k được tính bằng $C_1 \text{ OR } C_2 \text{ OR } \dots \text{ OR } C_k$

Cho hai đỉnh A và B , Tìm đường đi từ A đến B sao cho độ dài là nhỏ nhất? (một cạnh có thể đi qua nhiều lần)

Dữ liệu: Vào từ file văn bản **PATH.INP**

- Dòng 1: N, M ($1 \leq N \leq 10^3, 1 \leq M \leq 10^4$)
- M dòng tiếp theo mỗi dòng 3 số nguyên dương u, v, C_i ($1 \leq u, v \leq N, 1 \leq C_i \leq 1024$)
- Dòng cuối là hai số nguyên dương A (đỉnh bắt đầu) và B (đỉnh kết thúc)

Kết quả: Ghi ra file văn bản **PATH.OUT** Độ dài ngắn nhất (nếu không có in -1)

Ví dụ

PATH.INP	PATH.OUT	
3 4 1 2 1 1 2 1000 2 3 3 1 3 100 1 3	3	$1 \rightarrow 2 \rightarrow 3$ $C_{1,2} = 1$ $C_{2,3} = 3$ $1 \text{ OR } 3 = 3$

BÀI 71. SPECIAL WALK

Cho đồ thị có hướng không trọng số gồm N đỉnh và M cung. Một “đường đi đặc biệt” trên đồ thị là đường đi bắt đầu và kết thúc tại cùng một đỉnh u , hay là đường đi $u, a_1, a_2, \dots, a_i, \dots, u$ trong đó a_i đôi một khác nhau và $a_i \neq u$.

Yêu cầu: Với mỗi đỉnh của đồ thị, hãy xác định xem có tồn tại “đường đi đặc biệt” bắt đầu từ đỉnh đó không.

Dữ liệu: Vào từ file văn bản **SPECIALWALK.INP**

- Dòng 1: N, M ($1 \leq N \leq 10^5, 1 \leq M \leq 2 \times 10^5$)
- M dòng tiếp theo mỗi dòng 2 số nguyên dương u, v ($1 \leq u, v \leq N$) mô tả cung nối $u \rightarrow v$

Kết quả: Ghi ra file văn bản **SPECIALWALK.OUT** n số nguyên, số thứ i là 1 nếu tồn tại đường đi đặc biệt, là 0 nếu ngược lại

Ví dụ

SPECIALWALK.INP	SPECIALWALK.OUT	
5 5 1 2 2 3 3 4 4 5 4 2	0 1 1 1 0	$2 \rightarrow 3 \rightarrow 4 \rightarrow 2$

BÀI 72. Unique Trip

Đất nước HTland xinh đẹp có n thành phố và m con đường 1 chiều nối các thành phố. Các thành phố được chia thành các bang khác nhau. Với 2 thành phố A, B nếu tồn tại đường đi từ A đến B và quay về A thì 2 thành phố này thuộc cùng 1 bang.

- Nếu bạn di chuyển trên con đường nối 2 thành phố cùng một bang thì bạn mất phí là 0 đồng
- Nếu bạn di chuyển trên con đường nối 2 thành phố thuộc 2 bang khác nhau thì bạn mất phí là 1 đồng (đi qua Trạm thu giá)

Yêu cầu: Bạn hãy xác định số tiền tối thiểu để đảm bảo có thể di chuyển giữa hai thành phố bất kỳ và sử dụng bất kỳ con đường nào.

Dữ liệu: Vào từ file văn bản **UNIQUETRIP.INP**

- Dòng 1: N, M ($1 \leq N \leq 10^5, 1 \leq M \leq \min(2 \times 10^5, N * (N - 1))$)
- M dòng tiếp theo mỗi dòng 2 số nguyên dương u, v ($1 \leq u \neq v \leq N$) mô tả cung nối $u \rightarrow v$

Kết quả: Ghi ra file văn bản **UNIQUETRIP.OUT** một số nguyên duy nhất là số tiền tối thiểu phải chuẩn bị.

Ví dụ

UNIQUETRIP.INP	UNIQUETRIP.OUT
4 4 1 2 2 3 3 1 1 4	1

BÀI 73. MAGIC LABYRINTH

Mê cung gồm N phòng và M cửa ra vào. Cửa ra vào chỉ có thể đi theo một hướng, có cửa ra vào đi từ phòng này đến chính phòng đó, khi di chuyển qua một cửa ra vào, bạn sẽ nhận được 1 đồng tiền giá trị w .

Bạn có thể di chuyển qua các cửa bao nhiêu lần tùy ý nhưng mục tiêu của trò chơi là đạt được điểm số tối đa:

- Nếu bạn không sử dụng cửa ra vào, điểm của bạn là 0
- Gọi p là đồng tiền có giá trị cao nhất khi đi qua cửa ra vào, q là đồng tiền có giá trị nhỏ nhất khi qua cửa. Điểm của trò chơi là $p - q$

Yêu cầu: Xác định điểm tối đa mà bạn có thể nhận được.

Dữ liệu: Vào từ file văn bản **MAGICLAB.INP**

- Dòng 1: N, M ($2 \leq N \leq 3 \times 10^5, 1 \leq M \leq 3 \times 10^5$)
- M dòng tiếp theo mỗi dòng 3 số nguyên dương u, v, w ($1 \leq u, v \leq N$) mô tả cửa đi từ $u \rightarrow v$ và nhận được đồng tiền giá trị w ($0 \leq w \leq 10^9$)

Kết quả: Ghi ra file văn bản **MAGICLAB.OUT** một số nguyên duy nhất là số điểm tối đa nhận được.

Ví dụ

MAGICLAB.INP	MAGICLAB.OUT	Giải thích
6 8 1 2 9 2 3 7 3 1 1 2 4 5 3 4 4 4 5 6 5 6 3 6 4 10	9	$2 \rightarrow 3 \rightarrow 1 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 4$ Số điểm = $10 - 1 = 9$

BÀI 74. GCDGRAPH

Cho đồ thị có hướng N đỉnh M cung, các đỉnh đánh số từ 1 đến N và đỉnh i có trọng số là c_i . Chi phí một đường đi trên đồ thị là ước chung lớn nhất của tất cả các đỉnh trên

đường đi hay là đường đi v_1, v_2, \dots, v_k có chi phí $= GCD(C_{v_1}, C_{v_2}, \dots, C_{v_k})$. Chú ý rằng các v_i không nhất thiết phải đôi một khác nhau và $k \geq 1$.

Yêu cầu: Tìm đường đi có chi phí nhỏ nhất?

Dữ liệu: Vào từ file văn bản **GCDGRAPH.INP**

- Dòng 1: N, M ($1 \leq N, M \leq 10^5$)
- Dòng 2: N số nguyên dương c_i ($1 \leq c_i \leq 10^5$)
- M dòng tiếp theo mỗi dòng 3 số nguyên dương u, v, w ($1 \leq u, v \leq N$) mô tả cung đi từ $u \rightarrow v$.

Kết quả: Ghi ra file văn bản **GCDGRAPH.OUT** một số nguyên duy nhất là chi phí nhỏ nhất tìm được.

Ví dụ

GCDGRAPH.INP	GCDGRAPH.OUT	Giải thích
3 2 4 6 8 1 2 2 3	2	$1 \rightarrow 2 \rightarrow 3$ Chi phí = 2

BÀI 75. The transportation network

Đất nước HTland xinh đẹp có n thành phố, giữa hai thành phố có thể nối với nhau bằng đường tàu và đường bộ.

- Đường tàu và đường bộ đều là hai chiều.
- Hai thành phố u, v được gọi là “kết nối đường tàu” nếu tồn tại một đường đi qua các đường tàu giữa u và v
- Hai thành phố u, v được gọi là “kết nối đường bộ” nếu tồn tại một đường đi qua các đường bộ giữa u và v
- Hệ thống giao thông tại HTland được gọi là “cân bằng” nếu với mọi cặp thành phố u, v ta có: u, v “kết nối đường tàu” $\Leftrightarrow u, v$ “kết nối đường bộ”

Ban đầu, HTland chưa có bất kỳ 1 đường tàu hoặc đường bộ nào, bạn có q truy vấn cho biết sẽ xây dựng đường tàu hoặc đường bộ giữa hai thành phố u, v , sau mỗi truy vấn, bạn hãy cho biết hệ thống giao thông có “cân bằng” hay không?

Dữ liệu: Vào từ file văn bản **TRANSPORTATION.INP**

- Dòng 1: N, Q ($1 \leq N \leq 10^5, 1 \leq Q \leq 2 \times 10^5$)
- Q dòng tiếp theo mỗi dòng 3 số nguyên dương x, u, v ($1 \leq u, v \leq N$) mô tả truy vấn:
 - o 1 $u v$: xây dựng đường tàu giữa u, v
 - o 2 $u v$: xây dựng đường bộ giữa u, v

Kết quả: Ghi ra file văn bản **TRANSPORTATION.OUT** trên Q dòng, dòng thứ i là đáp án của truy vấn i , ghi **YES** nếu hệ thống giao thông là cân bằng, ngược lại ghi **NO**.

Ví dụ

TRANSPORTATION.INP	TRANSPORTATION.OUT
5 8	NO
1 1 2	NO
1 2 3	NO
2 1 3	YES
2 1 2	NO
1 3 4	NO
2 2 5	NO
1 4 5	YES
2 1 4	

BÀI 76. 999 đóa hồng

HD và bạn gái của anh ấy đang đi du lịch trên HTland, HTland là một quần đảo gồm n hòn đảo và m cây cầu nối các hòn đảo. Trên các cây cầu này có 1 số cây cầu trồng hoa hồng.

HD hiện đang ở hòn đảo x và bạn gái của anh ấy đang ở hòn đảo y (x có thể bằng y). HD muốn đi đến chỗ bạn gái của anh ấy và tặng cho bạn gái 1 đóa hoa hồng xinh đẹp vì thế anh ấy phải đi qua một cây cầu có trồng hoa hồng để hái trộm về tặng cho bạn gái.

Tuy nhiên:

- Các cây cầu rất yếu, sau khi HD đi qua cây cầu nào thì cây cầu ấy sụp đổ (không đi qua được lần nào nữa)
- HD cũng không thể đi qua nửa cầu và quay lại.

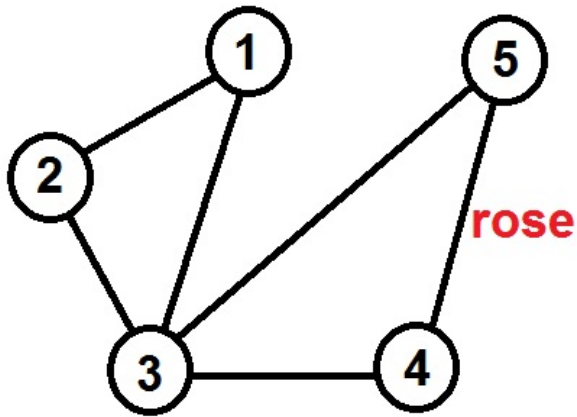
Yêu cầu: Bạn hãy cho biết HD có thể tặng cho bạn gái của mình 1 đóa hồng được không?

Dữ liệu: Vào từ file văn bản **ROSE.INP**

- Dòng 1: N, M ($1 \leq N \leq 3 \times 10^5, 1 \leq M \leq 3 \times 10^5$)
- M dòng tiếp theo mỗi dòng 3 số nguyên dương u, v, x ($1 \leq u, v \leq N$) mô tả cầu:
 - o $u v 1$: Cầu nối đảo $u v$ có hoa hồng
 - o $u v 0$: Cầu nối đảo $u v$ không có hoa hồng
- Dòng cuối cùng chứa hai số x, y

Kết quả: Ghi ra file văn bản **ROSE.OUT** ghi **YES** nếu HD tặng được hoa cho bạn gái, ngược lại ghi **NO**.

Ví dụ

ROSE.INP	ROSE.OUT	
5 6 1 2 0 2 3 0 3 1 0 3 4 0 4 5 1 5 3 0 1 2	YES	 <p>1 → 3 → 5 → 4 → 3 → 2</p>

BÀI 77. Journey Scheduling

Đất nước HTland xinh đẹp có n thành phố đánh số từ 1 đến n bố trí như một cây ($n - 1$ cạnh). HD quyết định đi du lịch trên đất nước này như sau:

- Xuất phát từ thành phố V
- Tham quan đúng K thành phố là kết thúc
- Từ thành phố hiện tại, đi thăm thành phố có khoảng cách xa nhất (theo đơn vị cạnh), nếu có nhiều thành phố có khoảng cách xa nhất bằng nhau thì:
 - o HD chọn thăm thành phố đã được tham quan với số lần ít nhất;
 - o Nếu số lần tham quan ít nhất bằng nhau thì tham quan thành phố có chỉ số nhỏ nhất

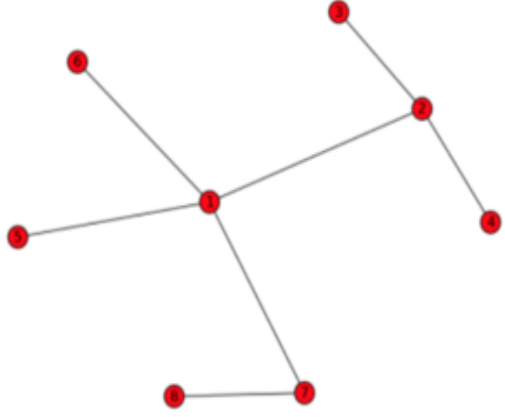
Yêu cầu: Tính tổng khoảng cách đi tham quan.

Dữ liệu: Vào từ file văn bản **JOURNEY1.INP**

- Dòng 1: N, M ($1 \leq N \leq 10^5, 1 \leq M \leq 10^5$) tương ứng là số đỉnh và số câu hỏi;
- $n - 1$ dòng tiếp theo mỗi dòng 2 số nguyên dương u, v ($1 \leq u, v \leq N$) mô tả đồ thị;
- m dòng tiếp theo, mỗi dòng chứa hai số V, K ($1 \leq K \leq 10^9$) tương ứng là câu hỏi

Kết quả: Ghi ra file văn bản **JOURNEY1.OUT** trên m dòng, dòng thứ i là tổng khoảng cách của câu hỏi.

Ví dụ

JOURNEY1.INP	JOURNEY1.OUT	
8 7	24	 <p>Test 4 – 6: $4 \rightarrow 8 \rightarrow 3 \rightarrow 8 \rightarrow 4 \rightarrow 8 \rightarrow 3$ Test 6 – 3: $6 \rightarrow 3[3,4,8] \rightarrow 8 \rightarrow 4$</p>
2 1	16	
3 2	11	
4 2	23	
5 1	24	
6 1	3	
7 1	23	
8 7		
4 6		
3 4		
6 3		
7 6		
4 6		
7 1		
2 6		

BÀI 78. Chocolate Journey

Tại HTland có N thành phố và M con đường hai chiều nối 2 thành phố (không có con đường nào nối 1 thành phố với chính nó và cũng không có hai con đường nào nối cùng hai thành phố) đánh số từ 1 đến M , con đường i có độ dài d_i và di chuyển trên con đường này mất d_i đơn vị thời gian.

HD đang sống tại thành phố B và bạn gái của Anh ấy sống tại thành phố A. HD muốn mang tặng cho bạn gái của mình một thanh socola, tuy nhiên socola ngon lại không có bán tại thành phố B và bán tại K thành phố khác.

Các thanh socola để trong tủ lạnh, khi HD mua thì thanh socola được bỏ ra khỏi tủ lạnh và chỉ được để ngoài trời trong x đơn vị thời gian.

Yêu cầu: Cho biết thời gian ngắn nhất để HD có thể mang thanh socola cho bạn gái?

Dữ liệu: Vào từ file văn bản **JOURNEY2.INP**

- Dòng 1: N, M, K ($1 \leq N \leq 3 \times 10^5, 1 \leq M \leq \min(10^6, N * (N - 1)/2), 1 \leq K \leq N - 1$)
- K dòng tiếp theo ghi chỉ số của các thành phố có bán socola
- M dòng tiếp theo mỗi dòng 3 số nguyên dương u, v, d ($1 \leq u, v \leq N$) mô tả đường nối thành phố $u - v$ có độ dài d ($1 \leq d \leq 500$)
- Dòng cuối cùng chứa hai số A, B

Kết quả: Ghi ra file văn bản **JOURNEY2.OUT** khoảng thời gian nhỏ nhất HD mang được socola cho bạn gái, nếu không mang được thì in -1

Ví dụ

JOURNEY2.INP	JOURNEY2.OUT
7 3 1 6 1 4 7 1 3 5 7 6 1 3 6 2	-1

BÀI 79. Buses Journey

Tại HTcity có n trạm xe bus đánh số từ 1 đến n . Có tất cả m xe bus, xe thứ i :

- Di chuyển lần lượt qua t_i trạm: $S_{i,1}, S_{i,2}, \dots, S_{i,t_i}$;
- Tại thời điểm 0 thì xe bus đỗ ở trạm $S_{i,1}$ và sau đó cứ 1 phút lại xe lại đến trạm tiếp theo.
- Khi đến trạm cuối cùng, xe sẽ di chuyển về trạm đầu tiên $S_{i,1}$ và tiếp tục hành trình.
- Các trạm giống nhau có thể xuất hiện nhiều lần trong hành trình của xe bus nhưng không liên kề với nhau (trạm cuối cùng tiếp giáp với trạm đầu tiên)
- Có thể có nhiều xe bus tại cùng 1 trạm trong cùng 1 thời điểm

Tại thời điểm 0, HD đang ở trạm số 1, tại một thời điểm bất kỳ:

- Nếu bạn và xe bus đang ở cùng 1 trạm, bạn có thể lên xe và di chuyển trên xe bus.
- Bạn có thể xuống xe bất kỳ trạm nào trong hành trình của xe bus.
- Bạn chỉ có thể sử dụng 1 xe bus tại 1 thời điểm
- Bạn chỉ có thể sử dụng nhiều xe bus để đến điểm kết thúc
- Bạn vào và xuống xe bus là tức thời (thời gian = 0)

Yêu cầu: Tìm thời gian tối thiểu để đến từng trạm?

Dữ liệu: Vào từ file văn bản **JOURNEY3.INP**

- Dòng 1: N, M ($1 \leq N \leq 3 \times 10^5, 1 \leq M \leq 10^5$)
- M dòng tiếp theo mỗi dòng bắt đầu bằng t_i ($t_i \geq 2$) và sau đó là t_i số nguyên dương $S_{i,1}, S_{i,2}, \dots, S_{i,t_i}$ ($1 \leq S_{i,j} \leq N, S_{i,j} \neq S_{i,j-1}, S_{i,n} \neq S_{i,1}$)

Biết rằng $t_1 + t_2 + \dots + t_n \leq 2 \times 10^5$

Kết quả: Ghi ra file văn bản **JOURNEY3.OUT** $n - 1$ số, số thứ i là thời gian ít nhất để đến trạm $i + 1$. Nếu không đến được ghi -1

Ví dụ

JOURNEY3.INP	JOURNEY3.OUT
8 4 2 5 4 3 6 1 2 4 4 2 1 3 2 7 8	2 3 4 6 3 -1 -1

BÀI 80. Grid

Cho bảng $N \times M$, bạn cần di chuyển từ ô $(1; 1)$ đến $(N; M)$. Ô $(i; j)$ có chiều cao là $A[i][j]$. Để thay đổi chiều cao của 1 ô bất kỳ từ x thành y bạn phải trả chi phí là $|x - y|$. Yêu cầu: Bạn hãy tính chi phí ít nhất để con đường từ ô $(1; 1)$ đến $(N; M)$ gồm các ô có chiều cao bằng nhau, biết rằng chỉ được di chuyển qua các ô kề cạnh.

Dữ liệu: Vào từ file văn bản **GRID.INP**

- Dòng 1: N, M ($1 \leq N, M \leq 100$)
- M dòng tiếp theo mỗi dòng chứa N số $A[i][j]$ ($1 \leq A[i][j] \leq 100$)

Kết quả: Ghi ra file văn bản **GRID.OUT** một số duy nhất là chi phí tối thiểu tìm được.

Ví dụ

GRID.INP	GRID.OUT	
5 5 1 1 1 1 1 9 9 9 9 1 1 3 3 3 1 1 9 9 9 9 1 1 1 1 1	6	Đường $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (1,4) \rightarrow (1,5) \rightarrow (2,5) \rightarrow (3,5) \rightarrow (3,4) \rightarrow (3,3) \rightarrow (3,2) \rightarrow (4,1) \rightarrow (5,1) \rightarrow (5,2) \rightarrow (5,3) \rightarrow (5,4) \rightarrow (5,5)$

BÀI 81. Đi ngược chiều

HTland có n thành phố đánh số từ 1 đến n và m con đường một chiều đánh số từ 1 đến m , con đường i được cho bởi:

- u_i, v_i, w_i có nghĩa là có con đường 1 chiều nối thành phố u_i với thành phố v_i , nếu bạn đi ngược chiều, tức là đi từ v_i đến u_i thì mất phí phạt là w_i . Đương nhiên nếu đi đúng chiều thì không mất phí phạt.

HD có q người bạn, người bạn i muốn di chuyển từ thành phố a_i đến thành phố b_i , chi phí của người i là $cost[i] =$ tổng số tiền phạt của người i .

Gọi

$$totalcost = \sum_{i=1}^q cost[i]$$

HD là chủ tịch thành phố, anh ấy có thể đổi chiều bao nhiêu con đường cũng được, vì vậy anh ấy sẽ đổi chiều một số con đường để sao cho $totalcost$ là nhỏ nhất?

Dữ liệu: Vào từ file văn bản **JOURNEY4.INP**

- Dòng 1: N, M ($1 \leq N, M \leq 5 \times 10^5$)
- M dòng tiếp theo mỗi dòng chứa 3 số nguyên dương u_i, v_i, w_i ($u_i \neq v_i, 1 \leq w_i \leq 10^6$)
- Dòng tiếp theo là số nguyên dương q ($1 \leq q \leq 5 \times 10^5$)
- q dòng tiếp theo, mỗi dòng 2 số nguyên dương a_i, b_i

Kết quả: Ghi ra file văn bản **JOURNEY4.OUT** một số nguyên duy nhất là số $totalcost$ ít nhất?

Ví dụ

JOURNEY4.INP	JOURNEY4.OUT	
7 7 4 6 7 4 5 10 5 7 3 3 5 3 1 2 5 7 3 6 2 6 1 2 6 5 7 4	10	Đổi chiều cạnh số 1 và cạnh số 3. $6 \rightarrow 5: cost = 0$ $7 \rightarrow 4: cost = 10$ $totalcost = 10$

