

MỤC LỤC

Bài 1. Thuật toán đổi chỗ trực tiếp.....	4
Bài 2. Thuật toán sắp xếp chọn	4
Bài 3. Thuật toán sắp xếp chèn.	5
Bài 4. Thuật toán sắp xếp nổi bọt.....	5
Bài 5. Thuật toán tìm kiếm tuyến tính	5
Bài 6 . Thuật toán tìm kiếm nhị phân.....	6
Bài 7. Vị trí đầu tiên	6
Bài 8. Vị trí cuối cùng	6
Bài 9. Đếm số lần xuất hiện của phần tử trong mảng sắp xếp.....	7
Bài 10. Quick Sort.....	8
Bài 11. Merge Sort	8
Bài 12. Số cặp bằng nhau.....	9
Bài 13. Khiêu vũ.....	10
Bài 14. Nhà gần nhất.....	10
Bài 15. Xếp gạch	11
Bài 16. Vắt sữa bò	11
Bài 17. Đổi chỗ.....	12
Bài 18. Sắp xếp chèn	12
Bài 19. (The 2014 ACM-ICPC Asia Jakarta Regional Contest) : Problems A ...	13
Bài 20. Ca sĩ Le Ro	14
Bài 21. In theo khuôn dạng	15
Bài 22. Counting sort	15
Bài 23. Cặp số có tổng bằng k.....	16
Bài 24. Cặp số có tổng nhỏ hơn k	16
Bài 25. Cặp số có tổng lớn hơn k.....	17
Bài 26. Tích của số lớn nhất và nhỏ nhất của 2 mảng	17
Bài 27. Hợp nhất 2 mảng	18
Bài 28. Điền số còn thiếu	18

Bài 29. Sắp xếp theo giá trị tuyệt đối.....	19
Bài 30. Hợp và giao của 2 mảng	20
Bài 31. Sắp xếp lại dãy con.....	20
Bài 32. Sắp xếp chữ số	21
Bài 32.2. Sắp xếp theo tần suất	21
Bài 33. Đổi chỗ ít nhất	22
Bài 34. Đếm cặp $x^y > y^x$	23
Bài 35. Giao của 3 dãy số.....	24
Bài 36. Sắp xếp chẵn lẻ	24
Bài 37. Biểu thức nhỏ nhất.....	25
Bài 38. Xếp hàng.....	26
Bài 39. Cặp phân tử có tổng gần 0 nhất	26
Bài 40. Số lặp đầu tiên trong mảng	27
Bài 41. Cặp số nguyên tố có tổng bằng N.....	28
Bài 42.1. Tìm cặp số có hiệu bằng X.....	28
Bài 42.2. Số nhỏ nhất lớn hơn $A[i]$	29
Bài 42.3. Số 0 ở cuối dãy	30
Bài 42.4. Sắp đặt lại dãy số	30
Bài 42.5. Số nhỏ nhất và nhỏ thứ 2	31
Bài 42.6. Số nhỏ hơn K	32
Bài 42.7. Sắp xếp chẵn lẻ 2	33
Bài 43. Vanya và đèn lồng	33
Bài 44. Dragons	34
Bài 45. BerSU Ball.....	35
Bài 46. A and B and Compilation Errors	36
Bài 47. Laptops	37
Bài 48. Sort the array.....	38
Bài 49. Two Teams Composing.....	39
Bài 50. Pashmak and Flower.....	40
Bài 51. Similar Pairs.....	41

Bài 52. Distinct Number	42
Bài 53. Apartment	43
Bài 54. Ferris Wheel.....	44
Bài 55. Concert Ticket	44
Bài 56. Restaurant Customer.....	45
Bài 57. Liên hoan phim 1	46
Bài 58. Tìm 2 số có tổng bằng x	47
Bài 59. Tìm 3 số có tổng bằng x	48
Bài 60. Tìm 4 số có tổng bằng x	48
Bài 61. Missing Coin Sum	49
Bài 62. Collecting Number	49
Bài 63. Đếm mảng con có tổng bằng x	50
Bài 64. Đếm mảng con có tổng bằng $x(2)$	51
Bài 65. Đếm mảng con chia hết cho K	51
Bài 66. Đếm mảng con có nhiều nhất k số khác nhau	52
Bài 67. Unique Subarray : Mảng con dài nhất mà mỗi phần tử chỉ xuất hiện 1 lần	53
Bài 68. Chia mảng thành k mảng con liên tiếp có tổng lớn nhỏ nhất.....	53
Bài 69.Sliding Median : Trung vị của cửa sổ	54
Bài 70. Kiểm tra đoạn lồng nhau	55
Bài 72. Factory Machine	57
Bài 73. Task and Deadline	57
Bài 74. Đếm đoạn lồng nhau.....	58
Bài 75. Liên hoan phim 2	59
Bài 76. Sliding Window Maximum	60

Mọi thắc mắc và góp ý về đề bài các bạn liên hệ với mình qua địa chỉ email:

andrew168545824@gmail.com hoặc Zalo/Telegram : 0965303260

Các bạn có thể tham khảo video lời giải của mình tại

<https://cutt.ly/WmI0f6O>

File bài tập này bao gồm các bài toán về thuật toán tìm kiếm và sắp xếp. Để học tốt phần này, các bạn cần nắm vững các thuật toán và có khả năng cài đặt các thuật toán này một cách thành thạo. Bên cạnh đó là biết sử dụng các hàm trong thư viện STL của C++ bao gồm : Hàm sort, stable_sort, upper_bound, lower_bound, binary_search và biết cách custom các hàm này để có thể ứng dụng nó linh hoạt vào các bài toán mà yêu cầu về sắp xếp và tìm kiếm phức tạp. Hoàn thành file bài tập này có thể giúp các bạn dễ dàng tiếp cận với các bài toán sắp xếp và tìm kiếm khác khó hơn.

SẮP XẾP, TÌM KIẾM, CỬA SỔ TRƯỢT

Bài 1. Thuật toán đổi chỗ trực tiếp

In ra các bước của thuật toán sắp xếp đổi chỗ trực tiếp

Ví dụ:

Input	Output
4	Buoc 1: 2 7 5 3
5 7 3 2	Buoc 2: 2 3 7 5
	Buoc 3: 2 3 5 7

Bài 2. Thuật toán sắp xếp chọn

In ra các bước của thuật toán sắp xếp chọn.

Ví dụ:

Input	Output
4	Buoc 1: 2 7 3 5
5 7 3 2	Buoc 2: 2 3 7 5

	Buoc 3: 2 3 5 7
--	-----------------

Bài 3. Thuật toán sắp xếp chèn.

In ra các bước của thuật toán sắp xếp chèn.

Ví dụ:

Input	Output
4	Buoc 0: 5
5 7 3 2	Buoc 1: 5 7
	Buoc 2: 3 5 7
	Buoc 3: 2 3 5 7

Bài 4. Thuật toán sắp xếp nổi bọt.

In ra các bước của thuật toán sắp xếp nổi bọt.

Ví dụ:

Input	Output
4	Buoc 1: 3 2 5 7
5 3 2 7	Buoc 2: 2 3 5 7

Bài 5. Thuật toán tìm kiếm tuyến tính

Kiểm tra xem phần tử x có nằm trong mảng hay không, nếu có in ra 1, ngược lại in ra 0

Ví dụ

Input	Output
5 3 1 2 3 4 5	1
6 4 1 1 2 0 7 8	0

Bài 6 . Thuật toán tìm kiếm nhị phân

Kiểm tra xem phần tử x có nằm trong mảng đã được sắp xếp hay không? Nếu có in ra 1, ngược lại in ra 0

Input	Output
5 3 1 2 3 4 5	1
6 4 0 1 1 2 7 8	0

Bài 7. Vị trí đầu tiên

Cho mảng số nguyên đã được sắp xếp tăng dần và số nguyên x. Tìm vị trí xuất hiện đầu tiên của x trong mảng hoặc xác định rằng nó không tồn tại.

Input

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng n và phần tử x; dòng tiếp theo là n số A [i] của mảng A []; các số được viết cách nhau một vài khoảng trống.

T, n thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^3$.

Output

In ra vị trí xuất hiện đầu tiên của x trong mảng, in ra -1 nếu x không nằm trong mảng

Ví dụ

Input	Output
2 5 3 1 2 3 3 3 5 4 1 1 2 5 6	3 -1

Bài 8. Vị trí cuối cùng

Cho mảng số nguyên đã được sắp xếp tăng dần và số nguyên x. Tìm vị trí xuất hiện cuối cùng của x trong mảng hoặc xác định rằng nó không tồn tại.

Input

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng n và phần tử x; dòng tiếp theo là n số A [i] của mảng A []; các số được viết cách nhau một vài khoảng trống.

T, n thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^3$.

Output

In ra vị trí xuất hiện cuối cùng của x trong mảng, in ra -1 nếu x không nằm trong mảng

Ví dụ

Input	Output
2 5 3 1 2 3 3 3 5 4 1 1 2 5 6	5 -1

Bài 9. Đếm số lần xuất hiện của phần tử trong mảng sắp xếp

Cho mảng số nguyên đã được sắp xếp tăng dần và số nguyên x. Đếm số lần xuất hiện của x trong mảng.

Input

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng n và phần tử x; dòng tiếp theo là n số A [i] của mảng A []; các số được viết cách nhau một vài khoảng trống.

T, n thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^3$.

Output

In ra số lần xuất hiện của x trong mảng

Ví dụ

Input	Output
2 5 3 1 2 3 3 3 5 4 1 1 2 5 6	3 0

Bài 10. Quick Sort

Để sắp xếp tăng dần một mảng A gồm n phần tử a_1, a_2, \dots, a_n , thuật toán sắp xếp nhanh (QuickSort) áp dụng phân hoạch để chia mảng A thành hai mảng con B và C sao cho $b_i \leq c_j$ (với mọi i, j). Có nhiều thuật toán phân hoạch, một trong số đó là thuật toán Lomuto. Thuật toán thực hiện như sau:

- Chọn phần tử cuối của mảng A làm chốt phân hoạch.
- Duyệt qua các phần tử của mảng A từ phần tử đầu đến phần tử kế chốt. Nếu phần tử nào nhỏ hơn hoặc bằng chốt thì hoán vị về đầu (đưa vào mảng B).
- Hoán vị chốt vào giữa sao cho chốt là phần tử phân định giữa B và C.

Ví dụ minh họa: phân hoạch 8 phần tử: 8 7 2 1 5 3 6 4. Chọn chốt phân hoạch là 4.

Hoán vị 2 về đầu: 2 7 8 1 5 3 6 [4]

Hoán vị 1 về đầu: 2 1 8 7 5 3 6 [4]

Hoán vị 3 về đầu: 2 1 3 7 5 8 6 [4]

Hoán vị chốt vào giữa: 2 1 3 [4] 5 8 6 7

(Các phần tử được gạch dưới ≤ 4 là mảng B, các phần tử còn lại ≥ 4 là mảng C)

Cho một mảng n phần tử bất kỳ, bạn hãy phân hoạch mảng trên dùng thuật toán Lomuto.

Input:

- Dòng đầu tiên là số nguyên n ($2 \leq n \leq 20$) là số phần tử của mảng.
- Dòng tiếp theo gồm n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq 100$), mỗi số cách nhau một khoảng trắng.

Output:

- Là một dòng gồm n phần tử sau khi đã phân hoạch, mỗi phần tử cách nhau một khoảng trắng. Phần tử chốt được đánh dấu bằng cặp dấu [].

Ví dụ

Input	Output
8 8 7 2 1 5 3 6 4	2 1 3 [4] 5 8 6 7

Source code tham khảo : <https://ideone.com/5TQhYK>

Source code thuật toán sắp xếp sử dụng phân hoạch Lomuto ở trên và phân hoạch Hoare. Phân hoạch Hoare tốt hơn so với Lomuto : <https://ideone.com/GPfz8T>

Bài 11. Merge Sort

Để sắp xếp tăng dần một mảng A gồm n phần tử a_1, a_2, \dots, a_n , thuật toán sắp xếp trộn (MergeSort) áp dụng chia đôi mảng A thành hai mảng B và C, sắp xếp B, C và sau đó trộn B và C cho ra mảng A tăng dần. Ví dụ minh họa phương pháp trộn:

- Mảng B gồm 4 phần tử b_1, b_2, b_3, b_4 đã sắp tăng dần: 1 2 4 6

- Mảng C gồm 4 phần tử c_1, c_2, c_3, c_4 đã sắp tăng dần: 3 5 8 9

Nếu trộn hai mảng trên theo dãy thứ tự trộn $b_1, b_2, c_1, b_3, c_2, b_4, c_3, c_4$ thì có

được mảng sắp là 1 2 3 4 5 6 8 9.

Cho một mảng B gồm n phần tử và mảng C gồm m phần tử. Hãy in ra dãy thứ tự trộn sao cho nếu áp dụng dãy thứ tự trộn trên thì mảng kết quả được sắp xếp tăng dần.

Input:

- Dòng đầu tiên là hai số nguyên n, m cách nhau một khoảng trắng ($1 \leq n, m \leq 20$) là số phần tử của mảng B và mảng C.
- Dòng thứ 2 gồm n số nguyên b_1, b_2, \dots, b_n ($1 \leq b_i \leq 100$), mỗi số cách nhau một khoảng trắng.
- Dòng thứ 3 gồm m số nguyên c_1, c_2, \dots, c_m ($1 \leq c_i \leq 100$), mỗi số cách nhau một khoảng trắng.

Output:

- Là dãy thứ tự trộn, mỗi phần tử cách nhau một khoảng trắng (xem thêm ví dụ để hiểu cách xuất). Nếu có nhiều dãy thứ tự trộn, chỉ cần in ra một dãy bất kỳ.

Ví dụ

Input	Output
4 4 1 2 4 6 3 5 8 9	b1 b2 c1 b3 c2 b4 c3 c4

Source code tham khảo : <https://ideone.com/23uGqp>

Bài 12. Số cặp bằng nhau

Cho một mảng gồm n số nguyên dương $a_1, a_2, a_3, \dots, a_n$. Hỏi có bao nhiêu cặp số bằng nhau? (Bao nhiêu cặp $a_i = a_j$ với $i \neq j$, (a_i, a_j) và (a_j, a_i) chỉ được tính là 1 cặp)

Input:

- Dòng thứ nhất là chiều dài n của mảng ($1 \leq n \leq 10^5$)
- Dòng thứ hai gồm n số nguyên $a_1, a_2, a_3, \dots, a_n$ ($1 \leq a_i \leq 10^5$), mỗi số cách nhau một khoảng trắng.

Output:

- Là số nguyên xác định số lượng các cặp bằng nhau, lưu ý là số lượng này có thể rất lớn nên sử dụng kiểu long long.

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

Source code tham khảo : <https://ideone.com/zi2XQo>

Bài 13. Khiêu vũ

Trong lớp học có n bạn nam và m bạn nữ. Các bạn nam có chiều cao là a_1, a_2, \dots, a_n . Các bạn nữ có chiều cao là b_1, b_2, \dots, b_m . Nhân dịp lễ tổng kết cuối năm, cả lớp dự định tổ chức buổi khiêu vũ nhưng có điều kiện là trong một đôi khiêu vũ bất kỳ, bạn nam phải cao hơn bạn nữ. Và mỗi bạn không tham gia quá 1 đôi khiêu vũ. Hãy tính số lượng cặp đôi nhiều nhất thỏa mãn yêu cầu trên.

Input: gồm 3 dòng

- Dòng thứ nhất là hai số n, m mỗi số cách nhau một khoảng trắng ($1 \leq n, m \leq 10^5$)
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n là chiều cao các bạn nam ($1 \leq a_i \leq 10^9$)
- Dòng thứ ba gồm m số nguyên b_1, b_2, \dots, b_m là chiều cao các bạn nữ ($1 \leq b_i \leq 10^9$)

Output:

- Số lượng đôi khiêu vũ nhiều nhất tính được.

Ví dụ

Input	Output
3 2 3 2 1 2 3	1
3 3 4 3 4 2 2 1	3

Source code tham khảo : <https://ideone.com/OQkAzK>

Bài 14. Nhà gần nhất

Trên một con đường mới mở đã xuất hiện lác đác một số căn nhà vừa xây xong. Người ta đánh địa chỉ bằng cách tính khoảng cách từ vị trí của căn nhà đến đầu đường theo đơn vị mét. Biết địa chỉ các căn nhà, hãy tìm khoảng cách giữa hai nhà gần nhau nhất.

Input: gồm 2 dòng

- Dòng thứ nhất là số nguyên n biểu thị số lượng các căn nhà ($2 \leq n \leq 10^5$)
- Dòng thứ hai gồm n số nguyên $a_1, a_2, a_3, \dots, a_n$, mỗi số cách nhau một khoảng trắng là địa chỉ của n căn nhà. ($0 \leq a_i \leq 10^9$). Dữ liệu cho đảm bảo không có 2 địa chỉ nào trùng nhau.

Output:

- Là số nguyên duy nhất cho biết khoảng cách giữa hai căn nhà gần nhau nhất.

Ví dụ

Input	Output
3 1 6 3	2

3 9 3 6	3
------------	---

Source code tham khảo : <https://ideone.com/HccloD>

Bài 15. Xếp gạch

Nam có n viên gạch được đánh số từ 1 đến n . Các viên gạch có độ cứng lần lượt là a_1, a_2, \dots, a_n . Một viên gạch có độ cứng x nghĩa là Nam có thể chồng lên trên viên gạch đó tối đa x viên gạch khác, nếu chồng nhiều hơn thì viên gạch đó bị vỡ. Hỏi Nam có thể sắp được chồng gạch cao nhất là bao nhiêu?

Input:

- Dòng đầu tiên là số nguyên n ($1 \leq n \leq 100$) - là số viên gạch.
- Dòng tiếp theo gồm n số nguyên a_1, a_2, \dots, a_n ($0 \leq a_i \leq 100$) mỗi số cách nhau một khoảng trắng.

Output:

- Là số nguyên xác định chiều cao cao nhất của chồng gạch mà Nam sắp được.

Ví dụ

Input	Output
4 1 2 3 4	4
4 0 0 0 0	1

Source code tham khảo : <https://ideone.com/2r2Ds9>

Bài 16. Vắt sữa bò

Vào một buổi sáng anh Bo sắp một đàn bò gồm n con bò để vắt sữa. Anh dự kiến là vào sáng hôm đó, con bò thứ i có khả năng sẽ vắt được a_i lít sữa. Tuy nhiên đàn bò của anh có đặc tính là cứ mỗi lần vắt sữa một con, những con còn lại trông thấy sợ quá nên sẽ bị giảm sản lượng mỗi con 01 lít sữa. Nếu vắt sữa con bò thứ nhất, $n-1$ con còn lại bị giảm sản lượng. Sau đó vắt sữa con bò thứ hai thì $n-2$ con còn lại bị giảm sản lượng.... Bạn hãy giúp anh Bo tính xem thứ tự vắt sữa bò như thế nào để số lượng sữa vắt được là nhiều nhất nhé.

Input: gồm 2 dòng

- Dòng thứ nhất là số nguyên n ($1 \leq n \leq 100$) là số lượng con bò.
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq 1000$) là sản lượng sữa của các con bò.

Output:

- Là một số nguyên xác định số lít sữa nhiều nhất mà anh Bo có thể vắt được.

Ví dụ

Input	Output
4 4 4 4 4	10
4 2 1 4 3	6

Source code tham khảo : <https://ideone.com/lGcTMf>

Bài 17. Đổi chỗ

Cho một dãy gồm n số nguyên $a[1], a[2], \dots, a[n]$ hãy đưa ra dãy các thao tác đổi chỗ để nhận được dãy không giảm.

Input:

- Dòng đầu chứa số nguyên $n \leq 10^5$
- Dòng thứ hai chứa n số nguyên $a[1], a[2], \dots, a[n]$ ($-10^9 \leq a[1], a[2], \dots, a[n] \leq 10^9$), các số cách nhau bởi dấu cách.

Output: Gồm một số dòng, mỗi dòng chứa hai số i, j ($1 \leq i, j \leq n; i \neq j$) mô tả một phép đổi chỗ phần tử ở vị trí thứ i với phần tử ở vị trí thứ j .

Ví dụ

Input	Output
5 5 4 3 2 1	1 5 2 4

Source code tham khảo : <https://ideone.com/jyA88m>

Bài 18. Sắp xếp chèn

Để sắp xếp tăng dần một mảng n phần tử a_0, a_1, \dots, a_{n-1} (chỉ số bắt đầu từ 0). thuật toán sắp xếp chèn thực hiện $n-1$ bước. Tại bước i , chèn phần tử a_i vào các phần tử a_0, a_1, \dots, a_{i-1} sao cho dãy kết quả a_0, a_1, \dots, a_i là tăng dần. Ví dụ minh họa:

Sắp xếp mảng 6 phần tử: 8 5 2 7 9 3

Sau bước 1: 5 8 2 7 9 3

Sau bước 2: 2 5 8 7 9 3

Sau bước 3: 2 5 7 8 9 3

Sau bước 4: 2 5 7 8 9 3

Sau bước 5: 2 3 5 7 8 9

(Tại bước 1, số 5 chèn vào vị trí 0; tại bước 2, số 2 được chèn vào vị trí 0; tiếp theo, số 7 được chèn vào vị trí 2; tiếp theo số 9 giữ nguyên vị trí 4, cuối cùng số 3 chèn vào vị trí 1)

Cho mảng n phần tử bất kỳ, bạn hãy cho biết tại mỗi bước thực hiện như trên

thì số nào được chèn vào vị trí nào nhé.

Input:

- Dòng đầu tiên là số nguyên n ($2 \leq n \leq 20$) là số phần tử của mảng.
- Dòng tiếp theo gồm n số nguyên a_0, a_1, \dots, a_{n-1} ($1 \leq a_i \leq 100$), mỗi số cách nhau một khoảng trắng.

Output: gồm $n - 1$ dòng thể hiện $n - 1$ bước

- Tại dòng i là 2 số nguyên a_i và k cách nhau một khoảng trắng. k là vị trí cần chèn của a_i

Ví dụ

Input	Output
6	5 0
8 5 2 7 9 3	2 0
	7 2
	9 4
	3 1

Source code : <https://ideone.com/bgsSac>

Bài 19. (The 2014 ACM-ICPC Asia Jakarta Regional Contest) : Problems A

Phân tích nhóm (phân nhóm, chia nhóm) là công việc phân chia các phần tử trong một tập hợp thành một hoặc nhiều nhóm mà trong đó, các phần tử trong cùng một nhóm sẽ giống nhau hơn so với phần tử thuộc nhóm khác. Cho một tập N số nguyên dương và một số nguyên dương K , nhiệm vụ của bạn là đếm xem có bao nhiêu nhóm. Biết rằng 2 phần tử được xếp chung nhóm với nhau nếu như chênh lệch giữa chúng không vượt quá K .

Ví dụ: với tập $N = 7$ số nguyên dương: 2, 6, 1, 7, 3, 4, 9 và $K = 1$ thì ta sẽ có các mối quan hệ sau:

- 2 và 1 chung một nhóm (chênh lệch giữa chúng là 1, không vượt quá K)
- 2 và 3 chung một nhóm
- 6 và 7 chung một nhóm
- 3 và 4 chung một nhóm

Vậy ta sẽ có 3 nhóm: {1, 2, 3, 4}, {6, 7} và {9}

Input: Dòng đầu tiên chứa số nguyên T - số bộ test cần kiểm tra ($T \leq 100$), mỗi bộ test gồm 2 dòng:

- Dòng đầu trong mỗi bộ test chứa 2 số nguyên dương N, K ($1 \leq N \leq 100$, $1 \leq K \leq 10^6$)
- Dòng thứ hai trong mỗi bộ test chứa N số nguyên dương - các phần tử của tập hợp (giá trị không vượt quá 106)

Output: gồm N dòng, mỗi dòng xuất theo mẫu sau "Case #X: Y", trong đó X là số thứ tự của bộ test (theo đúng thứ tự) và Y là kết quả cần tìm của bộ test đó

Ví dụ

Input	Output
4	Case #1: 3
7 1	Case #2: 1
2 6 1 7 3 4 9	Case #3: 2
7 2	Case #4: 8
2 6 1 7 3 4 9	
5 5	
15 1 20 4 17	
8 10	
100 200 300 400 500 600 700 800	

Source code tham khảo : <https://ideone.com/wnPT6j>

Bài 20. Ca sĩ Lê Ro

Ca sĩ nổi tiếng Lê Ro vừa nhận được các lời mời lưu diễn của n đoàn ca nhạc. Đoàn thứ i mời lưu diễn từ ngày ai đến ngày bi (ai, bi là các số nguyên, $ai \leq bi$). Tuy nhiên tại một thời điểm, Lê Ro chỉ có thể tham gia hát cho một đoàn duy nhất mà thôi.

Với mong muốn đem lời ca tiếng hát của mình đến nhiều khán giả nhất, Lê Ro quyết định sẽ chọn tham gia nhiều đoàn nhất có thể. Bạn hãy tính thử xem Lê Ro nên chọn tham gia những đoàn nào để số lượng đoàn là nhiều nhất mà không bị trùng nhau về mặt thời gian.

Input: gồm 02 dòng

- Dòng thứ nhất là số nguyên n là số đoàn ca nhạc ($1 \leq n \leq 1.000$)
- Trong n dòng tiếp theo, dòng thứ i gồm hai số ai, bi cách nhau một khoảng trắng ($1 \leq ai \leq bi \leq 10^9$) là ngày bắt đầu và ngày kết thúc lưu diễn của đoàn thứ i.

Output:

- Là số nguyên xác định số lượng đoàn nhiều nhất mà Lê Ro có thể tham gia.

Ví dụ

Input	Output
6 3 8 9 12 6 10 1 4 2 7 11 1 4	3
4 5 6 1 2 7 8 3 4	4

Source code tham khảo : <https://ideone.com/E8m9do>

Bài 21. In theo khuôn dạng

Cho mảng A[] gồm n số nguyên khác nhau. Hãy đưa ra các phần tử của mảng theo khuôn dạng lớn nhất, nhỏ nhất, lớn thứ hai, nhỏ thứ 2, ... Ví dụ với A[] = {9, 7, 12, 8, 6, 5} ta đưa ra : 12, 5, 9, 6, 8, 7.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng n; dòng tiếp theo là n số A[i] của mảng A[]; các số được viết cách nhau một vài khoảng trống.

T, n thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^3$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ

Input:	Output:
2	
7	
7 1 2 3 4 5 6	7 1 6 2 5 3 4
8	9 1 8 2 7 3 6 4
1 6 9 4 3 7 8 2	

Bài 22. Counting sort

Sắp xếp mảng gồm n số nguyên chỉ bao gồm các số 0, 1, 2.

Input

Dòng đầu tiên là số lượng test case T không quá 100.

Mỗi test case gồm 2 dòng, dòng đầu tiên là số lượng phần tử trong mảng ($1 \leq n \leq 1000$).

Dòng thứ 2 là các phần tử trong mảng

Output

In ra các phần tử trong dãy được sắp xếp tăng dần.

Ví dụ

Input	Output
2 5 1 1 0 0 2 6 1 1 1 0 0 2	 0 0 1 1 2 0 0 1 1 1 2

Bài 23. Cặp số có tổng bằng k

Cho mảng a gồm n phần tử và số nguyên dương k.

Đếm số lượng cặp số có tổng bằng k.

Input

Dòng thứ 1 là số lượng test case T ($1 \leq T \leq 100$).

Mỗi test case gồm 2 dòng, dòng thứ 1 là số lượng phần tử trong mảng và số nguyên dương k ($1 \leq n, k \leq 10^6$)

Dòng thứ 2 là n phần tử trong mảng ($-10^6 \leq a_i \leq 10^6$).

Output

In ra số lượng cặp số có tổng bằng k trên mỗi dòng.

Ví dụ

Input	Output
2 4 4 2 2 2 2 3 3 1 2 3	 6 1

Bài 24. Cặp số có tổng nhỏ hơn k

Cho mảng a gồm n phần tử và số nguyên dương k.

Đếm số lượng cặp số có tổng nhỏ hơn k.

Input

Dòng thứ 1 là số lượng test case T ($1 \leq T \leq 100$).

Mỗi test case gồm 2 dòng, dòng thứ 1 là số lượng phần tử trong mảng và số nguyên dương k ($1 \leq n, k \leq 10^6$)

Dòng thứ 2 là n phần tử trong mảng ($0 \leq a_i \leq 10^6$).

Output

In ra số lượng cặp số có tổng nhỏ hơn k trên mỗi dòng.

Ví dụ

Input	Output
2 4 5 2 2 2 2 3 4 1 2 4	6 1

Bài 25. Cặp số có tổng lớn hơn k

Cho mảng a gồm n phần tử và số nguyên dương k.

Đếm số lượng cặp số có tổng lớn hơn k.

Input

Dòng thứ 1 là số lượng test case T ($1 \leq T \leq 100$).

Mỗi test case gồm 2 dòng, dòng thứ 1 là số lượng phần tử trong mảng và số nguyên dương k ($1 \leq n, k \leq 10^6$)

Dòng thứ 2 là n phần tử trong mảng ($-10^6 \leq a_i \leq 10^6$).

Output

In ra số lượng cặp số có tổng lớn hơn k trên mỗi dòng.

Ví dụ

Input	Output
2 4 5 2 3 4 5 3 3 1 2 3	5 2

Bài 26. Tích của số lớn nhất và nhỏ nhất của 2 mảng

Cho mảng a gồm n phần tử và mảng b gồm m phần tử. Tìm tích giữa số lớn nhất trong mảng a và số nhỏ nhất trong mảng b.

Input

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

Mỗi test case gồm 3 dòng, dòng đầu tiên là n và m . ($1 \leq n, m \leq 10^6$)

Dòng thứ 2 là các phần tử trong mảng a . ($-10^9 \leq a_i \leq 10^9$)

Dòng thứ 3 là các phần tử trong mảng b . ($-10^9 \leq b_i \leq 10^9$)

Output

Mỗi test case in kết quả trên một dòng

Ví dụ

Input	Output
1 3 4 1 2 3 -2 3 4	-6

Bài 27. Hợp nhất 2 mảng

Cho mảng a gồm n phần tử và mảng b gồm m phần tử. Hợp nhất 2 mảng để được 1 mảng sắp xếp tăng dần.

Input

Dòng đầu tiên là số lượng test case T ($1 \leq T \leq 100$).

Mỗi test case gồm 3 dòng, dòng đầu tiên là n và m . ($1 \leq n, m \leq 10^6$)

Dòng thứ 2 là các phần tử trong mảng a . ($-10^9 \leq a_i \leq 10^9$)

Dòng thứ 3 là các phần tử trong mảng b . ($-10^9 \leq b_i \leq 10^9$)

Output

Mỗi test case in kết quả trên một dòng

Ví dụ

Input	Output
1 3 4 1 2 3 1 5 6 2	1 1 2 2 3 5 6

Bài 28. Điền số còn thiếu

Cho mảng $A[]$ gồm n số nguyên dương. Gọi L, R là min và max các phần tử của $A[]$. Nhiệm vụ của bạn là tìm số phần tử cần thiết cần thêm vào mảng để mảng có đầy đủ các số trong

khoảng $[L, R]$. Ví dụ $A[] = \{5, 7, 9, 3, 6, 2\}$ ta nhận được kết quả là 2 tương ứng với các số còn thiếu là 4, 8.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n , tương ứng với số phần tử của mảng $A[]$; dòng tiếp theo là n số $A[i]$; các số được viết cách nhau một vài khoảng trống.

$T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n, A[i] \leq 10^6$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1
5	0
4 5 3 8 6	
3	
2 1 3	

Bài 29. Sắp xếp theo giá trị tuyệt đối

Cho mảng $A[]$ gồm n phần tử và số X . Hãy đưa sắp xếp các phần tử của mảng theo trị tuyệt đối của $|X - A[i]|$. Ví dụ với $A[] = \{10, 5, 3, 9, 2\}$ và $X = 7$ ta đưa ra mảng được sắp xếp theo nguyên tắc kể trên: $A[] = \{5, 9, 10, 3, 2\}$ vì $|7-10|=3, |7-5|=2, |7-3|=4, |7-9|=2, |7-2|=5$. Trong trường hợp có nhiều phần tử có giá trị tuyệt đối như nhau, ưu tiên theo thứ tự số xuất hiện trước trong mảng ban đầu.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng n và X ; dòng tiếp theo là n số $A[i]$ của mảng $A[]$; các số được viết cách nhau một vài khoảng trống.

T, n, X thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n, X, A[i] \leq 10^5$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	5 9 10 3 2
5 7	5 4 3 2 1
10 5 3 9 2	
5 6	
1 2 3 4 5	

Bài 30. Hợp và giao của 2 mảng

Cho 2 mảng đã được sắp xếp tăng dần, thực hiện tìm hợp và giao của 2 mảng. Các phần tử trong mỗi mảng khác nhau đôi một.

Input

Dòng đầu tiên là số lượng phần tử của 2 dãy n và m . ($1 \leq n, m \leq 10^6$).

Dòng thứ 2 là n phần tử trong dãy số 1. ($-10^6 \leq a_i \leq 10^6$).

Dòng thứ 3 là m phần tử trong dãy thứ 2. ($-10^6 \leq a_i \leq 10^6$).

Output

Dòng thứ 1 là hợp của 2 mảng

Dòng thứ 2 là giao của 2 mảng

Input	Output
4 5	
1 2 3	1 2 3 5 9
1 2 3 5 9	1 2 3

Bài 31. Sắp xếp lại dãy con

Cho mảng $A[]$ gồm n phần tử. Hãy tìm dãy con liên tục của mảng $A[R], \dots, A[L]$ sao cho khi sắp xếp lại dãy con ta nhận được một mảng được sắp xếp. Ví dụ với $A[] = \{10, 12, 20, 30, 25, 40, 32, 31, 35, 50, 60\}$ ta chỉ cần sắp xếp lại dãy con từ $A[4], \dots, A[9]: \{30, 25, 40, 32, 31, 35\}$ để có mảng được sắp.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng $A[]$; dòng tiếp theo là n số $A[i]$ của mảng $A[]$ các số được viết cách nhau một vài khoảng trống.

$T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100; 1 \leq n \leq 10^6; 0 \leq A[i] \leq 10^7$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	4 9
11	3 6
10 12 20 30 25 40 32 31 35 50 60	
9	
0 1 15 25 6 7 30 40 50	

Code tham khảo : <https://ideone.com/CaBWRV>

Bài 32. Sắp xếp chữ số

Cho mảng $A[]$ gồm n phần tử. Nhiệm vụ của bạn là đưa ra mảng đã được sắp xếp bao gồm các chữ số của mỗi phần tử trong $A[]$. Ví dụ $A[] = \{110, 111, 112, 113, 114\}$ ta có kết quả là $\{0, 1, 2, 3, 4\}$.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng $A[]$; dòng tiếp theo là n số $A[i]$; các số được viết cách nhau một vài khoảng trống.

$T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^7$; $0 \leq A[i] \leq 10^{16}$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1 3 4 8
3	1 2 3 4 6
131 11 48	
4	
111 222 333 446	

Bài 32.2. Sắp xếp theo tần suất

Cho mảng $A[]$ gồm n số nguyên. Nhiệm vụ của bạn là sắp xếp mảng theo số lần xuất hiện các phần tử của mảng. Số xuất hiện nhiều lần nhất đứng trước. Nếu hai phần tử có số lần

xuất hiện như nhau, số nhỏ hơn đứng trước. Ví dụ $A[] = \{5, 5, 4, 6, 4\}$, ta nhận được kết quả là $A[] = \{4, 4, 5, 5, 6\}$.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n , tương ứng với số phần tử của mảng $A[]$ và số k ; dòng tiếp theo là n số $A[i]$; các số được viết cách nhau một vài khoảng trống.

$T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^4$; $1 \leq k \leq 10^3$; $1 \leq A[i] \leq 10^5$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	4 4 5 5 6
5	9 9 9 2 5
5 5 4 6 4	
5	
9 9 9 2 5	

Code tham khảo : <https://ideone.com/7WYN3j>

Bài 33. Đổi chỗ ít nhất

Cho mảng $A[]$ gồm n phần tử. Hãy tìm số phép đổi chỗ ít nhất giữa các phần tử của mảng để mảng $A[]$ được sắp xếp. Ví dụ với $A[] = \{4, 3, 2, 1\}$ ta cần thực hiện ít nhất 2 phép đổi chỗ: $\text{Swap}(A[0], A[3])$, $\text{Swap}(A[1], A[2])$.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên là số phần tử của mảng n và X ; dòng tiếp theo là n số $A[i]$ của mảng $A[]$; các số được viết cách nhau một vài khoảng trống.

T, n thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^3$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	2
4	2
4 3 2 1	
5	
1 5 4 3 2	

Code tham khảo : <https://ideone.com/0mEFYG>

Bài 34. Đếm cặp $x^y > y^x$

Cho mảng $X[]$ gồm n phần tử và mảng $Y[]$ gồm m phần tử. Hãy đếm số các cặp $x^y > y^x$, trong đó $x \in X[]$ và $y \in Y[]$. Ví dụ $X[] = \{2, 1, 6\}$, $Y[] = \{1, 5\}$ ta có kết quả là 3 cặp (2, 1), (2, 5), (6, 1).

Input:

Dòng đầu tiên đưa vào số lượng bộ test T .

Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm ba dòng: dòng đầu tiên đưa vào n , m tương ứng với số phần tử của mảng $X[]$ và $Y[]$; dòng tiếp theo là n số $X[i]$ của mảng $X[]$; dòng cuối cùng là m số của mảng $Y[]$; các số được viết cách nhau một vài khoảng trống.

$T, n, m, X[i], Y[j]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n, m \leq 10^5$; $1 \leq X[i], Y[j] \leq 10^3$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
1	3
3 2	
2 1 6	
1 5	

Source code tham khảo:

<https://ideone.com/Xsm9NH>

Bài 35. Giao của 3 dãy số

Cho ba dãy số A[], B[], C[] gồm N1, N2, N3 phần tử đã được sắp xếp. Hãy đưa ra các phần tử có mặt trong cả ba dãy theo thứ tự tăng dần. Nếu không có đáp án, in ra -1.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm bốn dòng: dòng thứ nhất đưa vào N1, N2, N3 là số phần tử của mảng A[], B[], C[]; các dòng tiếp theo đưa vào 3 dãy A[], B[], C[].

Ràng buộc: $1 \leq T \leq 100$; $1 \leq N1, N2, N3 \leq 10^6$, $0 \leq A[i], B[j], C[k] \leq 10^{18}$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Ví dụ:

Input:	Output:
1	20 80
6 5 8	
1 5 10 20 40 80	
6 7 20 80 100	
3 4 15 20 30 70 80 120	

Code tham khảo : <https://ideone.com/Iy4iMC>

Bài 36. Sắp xếp chẵn lẻ

Cho dãy số A[] có n phần tử. Hãy sắp xếp các số chẵn trong dãy theo thứ tự tăng dần và các số lẻ theo thứ tự giảm dần.

In ra dãy kết quả đã sắp xếp trong đó vị trí số chẵn và vị trí số lẻ không thay đổi so với dãy ban đầu.

Input

Dòng đầu ghi số n ($1 < n \leq 1000$)

Các dòng tiếp theo ghi đủ n số của dãy A[], các số đều nguyên dương và không quá 1000.

Output

Ghi ra dãy kết quả đã sắp xếp trong đó các vị trí của số chẵn và số lẻ không thay đổi.

Ví dụ

Input	Output
-------	--------

10 1 2 3 4 5 6 7 7 9 6	9 2 7 4 7 6 5 3 1 6
---------------------------	---------------------

Code tham khảo : <https://ideone.com/zhYoUG>

Bài 37. Biểu thức nhỏ nhất

Một dãy gồm n số nguyên không âm a_1, a_2, \dots, a_n được viết thành một hàng ngang, giữa hai số liên tiếp có một khoảng trắng, như vậy có tất cả $(n-1)$ khoảng trắng. Người ta muốn đặt k dấu cộng và $(n-1-k)$ dấu trừ vào $(n-1)$ khoảng trắng đó để nhận được một biểu thức có giá trị lớn nhất.

Ví dụ, với dãy gồm 5 số nguyên 28, 9, 5, 1, 69 và $k = 2$ thì cách đặt $28+9-5-1+69$ là biểu thức có giá trị lớn nhất.

Yêu cầu: Cho dãy gồm n số nguyên không âm a_1, a_2, \dots, a_n và số nguyên dương k , hãy tìm cách đặt k dấu cộng và $(n-1-k)$ dấu trừ vào $(n-1)$ khoảng trắng để nhận được một biểu thức có giá trị lớn nhất.

Input

- Dòng đầu chứa hai số nguyên dương n, k ($k < n \leq 10^5$);
- Dòng thứ hai chứa n số nguyên không âm a_1, a_2, \dots, a_n ($a_n \leq 10^6$)

Output

- Một số nguyên là giá trị của biểu thức đạt được.

Ví dụ

Input:	Output:
Test 1	Test 1
5 2	100
28 9 5 1 69	
Test 2	Test 2
5 1	9
10 20 15 5 1	

Chú ý: Bài này không cho phép thay đổi thứ tự các số trong biểu thức ban đầu.

Code tham khảo : <https://ideone.com/pQwTuW>

Bài 38. Xếp hàng

Tại sân bay, mọi người đang làm thủ tục để check in. Có tất cả N vị khách. Vị khách thứ i tới làm thủ tục tại thời điểm T[i] và cần D[i] thời gian để check in xong.

Các bạn hãy xác định xem thời điểm nào tất cả các vị khách làm xong thủ tục để lên máy bay?

Input

Dòng đầu tiên là số nguyên dương N ($N \leq 100$).

N dòng tiếp theo, mỗi dòng gồm 2 số nguyên cho biết thời điểm đến của vị khách thứ i và thời gian vị khách này làm xong thủ tục check in. Các giá trị này không vượt quá 10^6 .

Output

In ra đáp án tìm được.

Ví dụ:

Input	Output
3	15
2 1	
8 3	
5 7	

Code tham khảo : <https://ideone.com/8jFIbq>

Bài 39. Cặp phần tử có tổng gần 0 nhất

Cho mảng A[] gồm n phần tử, hãy tìm cặp phần tử có tổng gần nhất so với 0. Nếu có nhiều cặp có cùng tổng thì lấy cặp đầu tiên xuất hiện

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào n là số phần tử của mảng A[]; dòng tiếp theo đưa vào n số A[i]; các số được viết cách nhau một vài khoảng trống.

T, n, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $2 \leq N \leq 10^3$, $-10^6 \leq A[i] \leq 10^6$.

Output:

Đưa ra tổng gần nhất với 0 của cặp phần tử.

Input:	Output:
2	-68
3	-14
-8 -66 -60	
6	
-21 -67 -37 -18 4 -65	

Với $n = 10^3$, có thể dùng Brute Force ($O(n^2)$), cách tốt hơn là sắp xếp sau đó duy trì cặp có chỉ số nhỏ nhất.

Code tham khảo : <https://ideone.com/zRZBMC>

Bài 40. Số lặp đầu tiên trong mảng

Cho mảng A[] gồm N phần tử. Hãy tìm phần tử lặp lại đầu tiên của mảng. Ví dụ với mảng A[] = {5, 6, 1, 2, 1, 4} thì ta có 1 là phần tử đầu tiên lặp lại trong mảng. Nếu không tồn tại đáp án, in ra -1.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào số phần tử của mảng N; dòng tiếp theo là N số A[i] là các phần tử của mảng A[].

T, N, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^6$, $1 \leq A[i] \leq 10^6$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	-1
5	
1 2 3 4 5	30
6	
10 20 30 30 20 5 7	

Bài này có nhiều cách giải, với input $1 \leq A[i] \leq 10^6$ thì dùng mảng đánh dấu là tốt nhất, hoặc có thể dùng map, set.

Code tham khảo : <https://ideone.com/Zzq4Iv>

Bài 41. Cặp số nguyên tố có tổng bằng N

Số tự nhiên N. Hãy tìm cặp số nguyên tố đầu tiên có tổng là N (cặp số chứa số nhỏ hơn). Nếu không tồn tại cặp số nguyên tố có tổng bằng N, hãy đưa ra -1.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm là một số N được ghi trên một dòng.

T, N thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^6$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	2 2
4	
8	3 5

Code tham khảo : <https://ideone.com/pf3aig>

Bài 42.1. Tìm cặp số có hiệu bằng X

Bạn thử cài đặt sử dụng map, set, mảng đánh dấu, tìm kiếm nhị phân, brute force(TLE).

Cho mảng A[] gồm N phần tử và số X. Nhiệm vụ của bạn là tìm cặp phần tử $A[i] - A[j] = X$. Nếu tồn tại $A[i] - A[j] = X$ đưa ra 1, ngược lại đưa ra -1.

Input:

Dòng đầu tiên đưa vào số lượng bộ test T.

Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất là cặp số N, X; dòng tiếp theo là N số A[i] là các phần tử của mảng A[].

T, N, X, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N \leq 10^5$, $1 \leq X$, $A[i] \leq 10^5$.

Output:

Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1
6 78	
5 20 3 2 5 80	-1
5 45	
90 70 20 80 50	

Code tham khảo : <https://ideone.com/6PJteh>

Bài 42.2. Số nhỏ nhất lớn hơn A[i]

Cho mảng A[] gồm n phần tử. Nhiệm vụ của bạn là tìm giá trị nhỏ nhất lớn hơn A[i] ($i=0, 1, 2, \dots, n-1$). Đưa ra '_' nếu A[i] không có phần tử nhỏ hơn nó. Ví dụ với mảng A[] = {13, 6, 7, 12}, ta có kết quả là {_, 7, 12, 13}.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T.
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng A[]; dòng kế tiếp đưa vào n số A[i] của mảng; các số được viết cách nhau một vài khoảng trống.
- T, n, A[i] thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^6$; $10^6 \leq A[i] \leq 10^6$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	7 6 10 9 15 3 2 _ 8
9	_ 7 12 13
6 3 9 8 10 2 1 15 7	
4	
13 6 7 12	

Code tham khảo : <https://ideone.com/tAeWXI>

Bài 42.3. Số 0 ở cuối dãy

Cho mảng $A[]$ gồm n phần tử. Nhiệm vụ của bạn là hãy sắp đặt lại các phần tử của mảng sao cho các số 0 để ở cuối cùng, các phần tử khác không được bảo toàn thứ tự trước sau. Ví dụ với mảng $A[] = \{1, 2, 0, 0, 0, 3, 6\}$ ta có kết quả $A[] = \{1, 2, 3, 6, 0, 0, 0\}$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng $A[]$; dòng kế tiếp đưa vào n số $A[i]$ của mảng; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^7$; $0 \leq A[i] \leq 10^{18}$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1 2 3 6 0 0 0
7	1 2 3 0 0 0 0
1 2 0 0 0 3 6	
6	
0 1 0 2 0 3	

Code tham khảo: <https://ideone.com/cKiT1I>

Bài 42.4. Sắp đặt lại dãy số

Cho mảng $A[]$ gồm n phần tử. Nhiệm vụ của bạn là hãy sắp đặt lại các phần tử của mảng sao cho $A[i] = i$. Nếu phần tử $A[j]$ của có giá trị khác j , hãy đưa ghi vào -1 . Ví dụ với mảng $A[] = \{-1, -1, 6, 1, 9, 3, 2, -1, 4, -1\}$ ta có kết quả $A[] = \{-1, 1, 2, 3, 4, -1, 1, -1, -1, 9\}$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng $A[]$; dòng kế tiếp đưa vào n số $A[i]$ của mảng; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^7$; $1 \leq A[i] \leq 10^{18}$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	-1 1 2 3 4 -1 6 -1 -1 9
10	0 1 -1 3 -1 -1
-1 -1 6 1 9 3 2 -1 4 -1	
6	
0 -3 1 -2 3 -4	

Code tham khảo : Sử dụng map : <https://ideone.com/qIqnOf>

Bài 42.5. Số nhỏ nhất còn thiếu

Cho mảng $A[]$ gồm $n-1$ phần tử bao gồm các khác nhau từ 1, 2, ..., n. Hãy tìm số nguyên dương nhỏ nhất không có mặt trong mảng $A[]$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào n ; dòng tiếp theo đưa vào $n-1$ số $A[i]$; các số được viết cách nhau một vài khoảng trống.
- T, n, A thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq N$, $A[i] \leq 10^7$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	4
5	9
1 2 3 5	
10	
1 2 3 4 5 6 7 8 10	

Code tham khảo: <https://ideone.com/YG2U3D>

Bài 42.5. Số nhỏ nhất và nhỏ thứ 2

Cho mảng $A[]$ gồm n phần tử, hãy đưa ra số nhỏ nhất và số nhỏ thứ hai của mảng. Nếu không có số nhỏ thứ hai, hãy đưa ra -1.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .

- Những dòng kế tiếp đưa vào các bộ test. Mỗi bộ test gồm hai dòng: dòng thứ nhất đưa vào n là số phần tử của mảng $A[]$; dòng tiếp theo đưa vào n số $A[i]$; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^7$; $A[i] \leq 10^7$.

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1 2
10	-1
5 6 7 8 9 10 1 2 3 4	
5	
1 1 1 1 1	

Code tham khảo : <https://ideone.com/r8GXzS>

Bài 42.6. Số nhỏ hơn K

Cho mảng $A[]$ gồm n số nguyên dương và số k . Nhiệm vụ của bạn là hãy sắp đặt lại các phần tử của mảng sao cho các số nhỏ hơn hoặc bằng k đứng cạnh nhau. Ví dụ với mảng $A[] = \{2, 1, 5, 6, 3\}$, $k = 3$ ta chỉ cần thực hiện 1 phép đổi chỗ để có mảng $A[] = \{2, 1, 3, 6, 5\}$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng $A[]$ và số k ; dòng kế tiếp đưa vào n số $A[i]$ của mảng; các số được viết cách nhau một vài khoảng trống.
- $T, n, k, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^7$; $1 \leq A[i], k \leq 10^7$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1
5 3	2
2 1 5 6 3	
7 5	
2 7 9 5 8 7 4	

Code tham khảo: <https://ideone.com/RdQ6v9>

Bài 42.7. Sắp xếp chẵn lẻ 2

Cho mảng $A[]$ gồm n số nguyên dương. Nhiệm vụ của bạn là hãy sắp xếp lại các phần tử của mảng sao cho $A[i] \geq A[i-1]$ nếu i chẵn, $A[i] \leq A[i-1]$ nếu i lẻ. Ví dụ với mảng $A[] = \{1, 2, 2, 1\}$ ta được mảng được sắp $A[] = \{1, 2, 1, 2\}$.

Input:

- Dòng đầu tiên đưa vào số lượng bộ test T .
- Những dòng kế tiếp đưa vào T bộ test. Mỗi bộ test gồm hai dòng: dòng đầu tiên đưa vào n là số phần tử của mảng $A[]$; dòng kế tiếp đưa vào n số $A[i]$ của mảng; các số được viết cách nhau một vài khoảng trống.
- $T, n, A[i]$ thỏa mãn ràng buộc: $1 \leq T \leq 100$; $1 \leq n \leq 10^3$; $1 \leq A[i] \leq 10^3$;

Output:

- Đưa ra kết quả mỗi test theo từng dòng.

Input:	Output:
2	1 2 1 2
4	1 3 2
1 2 2 1	
3	
1 3 2	

Code tham khảo :<https://ideone.com/1suXmG>

Bài 43. Vanya và đèn lồng

Vanya đi bộ vào ban đêm dọc theo một con đường thẳng dài có độ dài l , được thắp sáng bởi n chiếc đèn lồng. Xét hệ trục tọa độ với điểm đầu của đường phố tương ứng với điểm 0 và điểm cuối của nó tương ứng với điểm l . Khi đó đèn lồng thứ i ở điểm a_i . Đèn lồng chiếu sáng tất cả các điểm trên đường phố cách nó nhiều nhất là d , trong đó d là một số dương, chung cho tất cả các đèn lồng.

Vanya tự hỏi: bán kính ánh sáng tối thiểu d mà những chiếc đèn lồng phải có để thắp sáng cả con phố?

Input

Dòng đầu tiên chứa hai số nguyên n, l ($1 \leq n \leq 1000$, $1 \leq l \leq 10^9$) - số lượng đèn lồng và chiều dài đường phố tương ứng.

Dòng tiếp theo chứa n số nguyên a_i ($0 \leq a_i \leq 1$). Nhiều đèn lồng có thể được đặt tại cùng một điểm. Đèn lồng có thể nằm ở cuối phố.

Output

In bán kính ánh sáng tối thiểu d , cần thiết để chiếu sáng cả đường phố. Câu trả lời sẽ được coi là đúng nếu sai số tuyệt đối hoặc tương đối của nó không vượt quá 10^{-9}

Ví dụ

Input	Output
7 15 15 5 3 7 9 14 0	2.5000000000
2 5 2 5	2.0000000000

Link submit : <https://codeforces.com/problemset/problem/492/B>

Code tham khảo: <https://ideone.com/06VgDD>

Bài 44. Dragons

Kirito đang bị mắc kẹt ở cấp độ của MMORPG mà anh ấy đang chơi hiện tại. Để tiếp tục trò chơi, anh ta phải đánh bại tất cả n con rồng sống ở cấp độ này. Kirito và những con rồng có sức mạnh, được biểu thị bằng một số nguyên. Trong cuộc đọ sức giữa hai đối thủ, kết quả của cuộc đọ sức được quyết định bởi sức mạnh của họ. Ban đầu, sức mạnh của Kirito bằng s .

Nếu Kirito bắt đầu đấu tay đôi với rồng thứ i ($1 \leq i \leq n$) và sức mạnh của Kirito không lớn hơn sức mạnh của rồng có sức mạnh là x_i , thì Kirito thua trận đấu và chết. Nhưng nếu sức mạnh của Kirito lớn hơn sức mạnh của con rồng, thì anh ta sẽ đánh bại con rồng và được tăng thêm sức mạnh theo là y_i .

Kirito có thể chiến đấu với những con rồng theo bất kỳ thứ tự nào. Xác định xem liệu anh ta có thể chuyển sang cấp độ tiếp theo của trò chơi hay không, tức là đánh bại tất cả những con rồng mà không bị thua một lần nào.

Input

Dòng đầu tiên chứa hai số nguyên được phân tách bằng dấu cách s và n ($1 \leq s \leq 10^4$, $1 \leq n \leq 10^3$). Sau đó n dòng tiếp theo: dòng thứ i chứa các số nguyên được phân tách bằng dấu cách là x_i và y_i ($1 \leq x_i \leq 10^4$, $0 \leq y_i \leq 10^4$) - sức mạnh của con rồng thứ i và sức mạnh được tăng thêm khi đánh bại nó.

Output

Trên một dòng duy nhất in "YES" (không có dấu ngoặc kép), nếu Kirito có thể chuyển sang cấp độ tiếp theo và in "NO" (không có dấu ngoặc kép), nếu anh ta không thể.

Ví dụ

Input	Output
2 2 1 99 100 0	YES
10 1 100 100	NO

Link submit : <https://codeforces.com/problemset/problem/230/A>

Code tham khảo : <https://ideone.com/yLl8nU>

Bài 45. BerSU Ball

Đại học Bang Berland đang tổ chức một buổi khiêu vũ trong lễ kỷ niệm 100500 năm thành lập! n các chàng trai và m cô gái đã bận rộn luyện tập các động tác nhảy múa.

Chúng tôi biết rằng một số cặp nam và nữ sẽ được mời tham dự vũ hội. Tuy nhiên, kỹ năng khiêu vũ của các đôi tác trong mỗi cặp khác nhau nhiều nhất là một đơn vị.

Đối với mỗi cậu bé, chúng tôi biết kỹ năng nhảy của cậu ấy. Tương tự, đối với mỗi cô gái, chúng tôi biết kỹ năng khiêu vũ của cô ấy. Viết mã xác định số cặp lớn nhất có thể được hình thành từ n trai và m gái.

Input

Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 100$) - số bé trai. Dòng thứ hai chứa dãy a_1, a_2, \dots, a_n ($1 \leq a_i \leq 100$), trong đó a_i là kỹ năng nhảy của cậu bé thứ i.

Tương tự, dòng thứ ba chứa số nguyên m ($1 \leq m \leq 100$) - số bé gái. Dòng thứ tư chứa dãy b_1, b_2, \dots, b_m ($1 \leq b_j \leq 100$), trong đó b_j là kỹ năng nhảy của cô gái thứ j.

Output

In một số duy nhất - số cặp tối đa được yêu cầu.

Input	Output
4 1 4 6 2 5 5 1 5 7 9	3
4 1 2 3 4	0

4 10 11 12 13	
5 1 1 1 1 1 3 1 2 3	2

Link submit : <https://codeforces.com/problemset/problem/489/B>

Code tham khảo : <https://ideone.com/AlZypL>

Bài 46. A and B and Compilation Errors

A và B đang chuẩn bị cho các cuộc thi lập trình.

B rất thích gỡ lỗi code của mình. Nhưng trước khi chạy giải pháp và bắt đầu gỡ lỗi, trước tiên anh ta phải biên dịch mã.

Ban đầu, trình biên dịch hiển thị n lỗi biên dịch, mỗi lỗi được biểu diễn dưới dạng số nguyên dương. Sau một số nỗ lực, B đã sửa chữa được một số sai lầm và sau đó là một sai lầm khác.

Tuy nhiên, mặc dù B chắc chắn rằng anh ta đã sửa hai lỗi, anh ta không thể hiểu chính xác lỗi biên dịch nào đã biến mất - trình biên dịch của ngôn ngữ mà B sử dụng luôn hiển thị lỗi theo thứ tự mới! B chắc chắn rằng không giống như nhiều ngôn ngữ lập trình khác, các lỗi biên dịch đối với ngôn ngữ lập trình của mình không phụ thuộc vào nhau, tức là nếu bạn sửa một lỗi thì tập hợp các lỗi khác không thay đổi.

Bạn có thể giúp B tìm ra chính xác hai lỗi mà anh ấy đã sửa không?

Input

Dòng đầu tiên của đầu vào chứa số nguyên n ($3 \leq n \leq 10^5$) - số lỗi biên dịch ban đầu.

Dòng thứ hai chứa n số nguyên được phân tách bằng dấu cách a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) - các lỗi mà trình biên dịch hiển thị lần đầu tiên.

Dòng thứ ba chứa $n - 1$ số nguyên được phân tách bằng dấu cách b_1, b_2, \dots, b_{n-1} - các lỗi hiển thị ở lần biên dịch thứ hai. Đảm bảo rằng chuỗi ở dòng thứ ba chứa tất cả các số của chuỗi thứ hai ngoại trừ chính xác một số.

Dòng thứ tư chứa $n - 2$ số nguyên được phân tách bằng dấu cách c_1, c_2, \dots, c_{n-2} - các lỗi được hiển thị ở lần tổng hợp thứ ba. Đảm bảo rằng dãy ở dòng thứ tư chứa tất cả các số của dòng thứ ba ngoại trừ chính xác một số.

Output

In hai số trên một dòng: số lỗi biên dịch lần lượt biến mất sau khi B thực hiện lần sửa đầu tiên và lần thứ hai.

Ví dụ

Input	Output
5 1 5 8 123 7 123 7 5 1 5 1 7	8 123
6 1 4 3 3 5 7 3 7 5 4 3 4 3 7 5	1 3

Giải thích Test 1 : Sau lần sửa lỗi đầu tiên, lỗi được sửa là lỗi 8, sau lần sửa thứ 2 lỗi biến mất là lỗi 123.

Link Submit : <https://codeforces.com/problemset/problem/519/B>

Code tham khảo : <https://ideone.com/3A98pQ>

Bài 47. Laptops

Một ngày nọ, Dima và Alex có một cuộc tranh cãi về giá cả và chất lượng của máy tính xách tay. Dima cho rằng máy tính xách tay càng đắt tiền thì càng tốt. Alex không đồng ý. Alex cho rằng có hai máy tính xách tay, như vậy giá của máy tính xách tay đầu tiên thấp hơn (nhỏ hơn) so với giá của máy tính xách tay thứ hai nhưng chất lượng của máy tính xách tay thứ nhất cao hơn (lớn hơn) so với chất lượng của máy tính xách tay thứ hai.

Vui lòng kiểm tra phỏng đoán của Alex. Bạn được cung cấp mô tả của n máy tính xách tay. Xác định xem hai máy tính xách tay được mô tả ở trên có tồn tại không.

Input

Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 10^5$) - số lượng máy tính xách tay.

N dòng tiếp theo chứa hai số nguyên, ai và bi ($1 \leq ai, bi \leq n$), trong đó ai là giá của máy tính xách tay thứ i và bi là số đại diện cho chất lượng của máy tính xách tay thứ i (số lượng càng lớn thì chất lượng càng cao).

Tất cả các ai đều khác biệt. Tất cả các bi đều khác biệt.

Output

Nếu Alex đúng, hãy in "Happy Alex", nếu không thì in "Poor Alex" (không có dấu ngoặc kép).

Ví dụ

Input	Output
-------	--------

2 1 2 2 1	Happy Alex
3 1 1 2 2 3 3	Poor Alex

Link submit : <https://codeforces.com/problemset/problem/456/A>

Code tham khảo : <https://ideone.com/yrIT9i>

Bài 48. Sort the array

Là một lập trình viên, bạn thích mảng rất nhiều. Vào ngày sinh nhật của bạn, bạn bè của bạn đã cho bạn một mảng gồm n số nguyên riêng biệt.

Thật không may, kích thước của a quá nhỏ. Bạn muốn một mảng lớn hơn! Bạn bè của bạn đồng ý cung cấp cho bạn một mảng lớn hơn, nhưng chỉ khi bạn có thể trả lời đúng câu hỏi sau: có thể sắp xếp mảng a (theo thứ tự tăng dần) bằng cách đảo ngược chính xác một đoạn của a không?

Input

Dòng đầu tiên của đầu vào chứa số nguyên n ($1 \leq n \leq 10^5$) - kích thước của mảng a .

Dòng thứ hai chứa n số nguyên phân biệt bằng dấu cách: $a[1], a[2], \dots, a[n]$ ($1 \leq a[i] \leq 10^9$).

Output

In "yes" hoặc "no" (không có dấu ngoặc kép), tùy thuộc vào câu trả lời.

Nếu câu trả lời của bạn là "yes", thì cũng in hai số nguyên được phân tách bằng dấu cách biểu thị chỉ số bắt đầu và kết thúc (bắt đầu không được lớn hơn kết thúc) của phân đoạn được đảo ngược. Nếu có nhiều cách chọn các chỉ số này, hãy in bất kỳ cách nào trong số chúng.

Ví dụ

Input	Output
3 3 2 1	yes 1 3
4 2 1 3 4	yes 1 2
4	No

3 1 2 4	
2	yes
1 2	1 1

Link submit : <https://codeforces.com/problemset/problem/451/B>

Code tham khảo : <https://ideone.com/v0CdMp>

Bài 49. Two Teams Composing

Bạn có n học sinh dưới sự kiểm soát của bạn và bạn phải soạn chính xác hai đội bao gồm một số tập hợp con của các học sinh của bạn. Mỗi học sinh có một kỹ năng riêng, kỹ năng của học sinh thứ i được biểu thị bằng một số nguyên a_i (các học sinh khác nhau có thể có các kỹ năng giống nhau).

Vì vậy, về các đội. Thứ nhất, hai đội này nên có cùng số lượng thành viên. Hai ràng buộc nữa:

Đội đầu tiên phải bao gồm những học sinh có các kỹ năng riêng biệt (nghĩa là tất cả các kỹ năng trong đội đầu tiên là khác nhau).

Đội thứ hai nên bao gồm những học sinh có cùng kỹ năng (tức là tất cả các kỹ năng của đội thứ hai đều như nhau).

Lưu ý rằng có thể cho phép một số học sinh của đội thứ nhất có kỹ năng giống như học sinh của đội thứ hai.

Hãy xem xét một số ví dụ (các kỹ năng được đưa ra):

[1,2,3], [4,4] không phải là một cặp đội tốt vì kích cỡ phải giống nhau;

[1,1,2], [3,3,3] không phải là một cặp đội tốt vì đội đầu tiên không nên chứa những học sinh có cùng kỹ năng;

[1,2,3], [3,4,4] không phải là một cặp đội tốt vì đội thứ hai nên chứa những học sinh có cùng kỹ năng;

[1,2,3], [3,3,3] là một cặp đội ăn ý;

[5], [6] là một cặp đội ăn ý.

Nhiệm vụ của bạn là tìm kích thước x tối đa có thể để có thể tạo thành một cặp đội hợp lệ, trong đó quy mô mỗi đội là x (các kỹ năng trong đội đầu tiên cần là duy nhất, các kỹ năng trong đội thứ hai phải giống nhau giữa họ). Một học sinh không thể là thành viên của nhiều hơn một nhóm.

Bạn phải trả lời t các trường hợp kiểm tra độc lập.

Input

Dòng đầu tiên của dữ liệu đầu vào chứa một số nguyên t là số lượng test case ($1 \leq t \leq 10^4$) - số lượng trường hợp thử nghiệm. Sau đó, t các trường hợp thử nghiệm theo sau.

Dòng đầu tiên của test case chứa một số nguyên n ($1 \leq n \leq 2 \cdot 10^5$) - số học sinh. Dòng thứ hai của test case chứa n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$), trong đó a_i là kỹ năng của học sinh thứ i . Các học sinh khác nhau có thể có những kỹ năng giống nhau.

Đảm bảo rằng tổng của n trên tất cả các trường hợp thử nghiệm không vượt quá $2 \cdot 10^5$ ($\sum n \leq 2 \cdot 10^5$).

Output

In ra kết quả mỗi test case trên 1 dòng.

Ví dụ

Input	Output
4	
7	3
4 2 4 1 4 3 4	1
5	0
2 1 5 4 3	2
1	
1	
4	
1 1 1 3	

Link submit : <https://codeforces.com/problemset/problem/1335/C>

Code tham khảo : <https://ideone.com/Xcdzkg>

Bài 50. Pashmak and Flower

Pashmak quyết định tặng Parmida một cặp hoa trong vườn. Có n bông hoa trong vườn và thứ i của chúng có số đẹp là b_i . Parmida là một cô gái rất kỳ lạ nên cô ấy không nhất thiết muốn có hai bông hoa đẹp nhất. Cô ấy muốn có những cặp hoa mà sự khác biệt về vẻ đẹp của chúng là tối đa có thể!

Nhiệm vụ của bạn là viết một chương trình tính toán hai thứ:

Vẻ đẹp khác biệt tối đa của hoa mà Pashmak có thể mang lại cho Parmida.

Số cách mà Pashmak có thể hái hoa. Hai cách được coi là khác nhau nếu và chỉ khi có ít nhất một bông hoa được chọn ở cách thứ nhất và không được chọn ở cách thứ hai.

Input

Dòng đầu tiên của đầu vào chứa n ($2 \leq n \leq 2 \cdot 10^5$). Trong dòng tiếp theo có n số nguyên cách nhau bởi dấu cách b_1, b_2, \dots, b_n ($1 \leq b_i \leq 10^9$).

Output

Dòng đầu ra duy nhất phải chứa hai số nguyên. Sự khác biệt về vẻ đẹp tối đa và số lượng cặp tương xứng.

Ví dụ

Input	Output
5 3 1 2 3 1	2 4

Link submit : <https://codeforces.com/problemset/problem/459/B>

Code tham khảo : <https://ideone.com/8wEtut>

Bài 51. Similar Pairs

Ta gọi hai số x và y là tương tự nếu chúng có cùng chẵn lẻ (cùng phần dư khi chia cho 2), hoặc nếu $|x - y| = 1$. Ví dụ, trong mỗi cặp (2,6), (4,3), (11,7), các số tương tự với nhau và trong các cặp (1,4), (3,12), họ không phải.

Bạn được cung cấp một mảng a gồm n (n chẵn) số nguyên dương. Kiểm tra xem có sự phân chia mảng như vậy thành từng cặp mà mỗi phần tử của mảng thuộc đúng một cặp hay không và các số trong mỗi cặp tương tự nhau.

Ví dụ, đối với mảng $a = [11, 14, 16, 12]$, có một phân hoạch thành các cặp (11,12) và (14,16). Các số trong cặp đầu tiên giống nhau vì chúng khác nhau một phần và trong cặp thứ hai vì chúng đều chẵn.

Input

Dòng đầu tiên chứa một số nguyên t ($1 \leq t \leq 1000$) - số test case.

Mỗi test case gồm 2 dòng :

Dòng đầu tiên chứa một số nguyên dương chẵn n ($2 \leq n \leq 50$) - độ dài của mảng a .

Dòng thứ hai chứa n số nguyên dương a_1, a_2, \dots, a_n ($1 \leq a_i \leq 100$).

Output

In YES nếu tìm được đáp án phân chia, ngược lại in NO

Ví dụ

Input	Output
7	
4	YES
11 14 16 12	NO
2	YES
1 8	YES
4	YES
1 1 1 1	YES
4	NO
1 2 5 6	
2	
12 13	
6	
1 6 3 10 5 8	
6	
1 12 3 10 5 8	

Link submit : <https://codeforces.com/problemset/problem/1360/C>

Code tham khảo : <https://ideone.com/WGd6bx>

Bài 52. Distinct Number

Bạn được cung cấp một danh sách gồm n số nguyên và nhiệm vụ của bạn là tính số giá trị khác biệt trong danh sách.

Input

Dòng nhập đầu tiên có số nguyên n : số giá trị.

Dòng thứ hai có n số nguyên x_1, x_2, \dots, x_n .

Output

In một số nguyên: số lượng các giá trị riêng biệt.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq x_i \leq 10^9$$

Ví dụ

Input	Output
5 1 2 3 3 2	3

5 1 2 3 4 5	5
5 1 1 1 1 1	1

Link submit : <https://cses.fi/problemset/task/1621/>

Code tham khảo : <https://ideone.com/ZZpjIr>

Bài 53. Apartment

Có n người nộp đơn và m căn hộ miễn phí. Nhiệm vụ của bạn là phân phối các căn hộ sao cho càng nhiều người đăng ký sẽ nhận được căn hộ càng tốt.

Mỗi người nộp đơn có một kích thước căn hộ mong muốn, và họ sẽ chấp nhận bất kỳ căn hộ nào có diện tích đủ gần với kích thước mong muốn.

Input

Dòng nhập đầu tiên có ba số nguyên n , m và k : số người đăng ký, số căn hộ và chênh lệch tối đa cho phép.

Dòng tiếp theo chứa n số nguyên a_1, a_2, \dots, a_n : diện tích căn hộ mong muốn của mỗi người đăng ký. Nếu kích thước mong muốn của người nộp đơn là x , người đó sẽ chấp nhận bất kỳ căn hộ nào có kích thước từ $x-k$ đến $x+k$.

Dòng cuối cùng ghi m số nguyên b_1, b_2, \dots, b_m : diện tích từng căn hộ.

Output

In một số nguyên: số người nộp đơn sẽ nhận được một căn hộ.

Ràng buộc

$$1 \leq n, m \leq 10^5$$

$$0 \leq k \leq 10^9$$

$$1 \leq a_i, b_i \leq 10^9$$

Ví dụ

Input	Output
4 3 5 60 45 80 60 30 60 75	2

Link submit: <https://cses.fi/problemset/task/1084/>

Code tham khảo : <https://ideone.com/rbk7X9>

Bài 54. Ferris Wheel

Có n đứa trẻ muốn đi đu quay, và nhiệm vụ của bạn là tìm một chiếc thuyền gondola cho mỗi đứa trẻ.

Mỗi chiếc gondola có thể có một hoặc hai người trong đó và ngoài ra, tổng trọng lượng của một chiếc gondola không được vượt quá x. Bạn biết cân nặng của mọi đứa trẻ.

Số lượng thuyền gondola tối thiểu cần thiết cho trẻ em là bao nhiêu?

Input

Dòng nhập đầu tiên chứa hai số nguyên n và x: số đứa trẻ và trọng lượng tối đa cho phép.

Dòng tiếp theo chứa n số nguyên p1, p2,..., pn: trọng lượng của mỗi đứa trẻ

Output

In một số nguyên: số lượng thuyền gondola tối thiểu.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq x \leq 10^9$$

$$1 \leq p_i \leq x$$

Ví dụ

Input	Output
4 10 7 2 3 9	3

Link submit: <https://cses.fi/problemset/task/1090/>

Code tham khảo: <https://ideone.com/yT5sQG>

Bài 55. Concert Ticket

Có n vé xem hòa nhạc có sẵn, mỗi vé có một mức giá nhất định. Sau đó, m khách hàng lần lượt đến.

Mỗi khách hàng thông báo mức giá tối đa mà họ sẵn sàng trả cho một vé và sau đó, họ sẽ nhận được một vé với mức giá gần nhất có thể sao cho không vượt quá mức giá tối đa.

Input

Dòng đầu tiên chứa các số nguyên n và m : số lượng vé và số lượng khách hàng.

Dòng tiếp theo ghi n số nguyên h_1, h_2, \dots, h_n : giá của từng vé.

Dòng cuối cùng chứa m số nguyên t_1, t_2, \dots, t_m : giá tối đa cho mỗi khách hàng theo thứ tự họ đến.

Output

In, cho mỗi khách hàng, giá mà họ sẽ trả cho vé của họ. Sau đó, vé không thể được mua lại lần nữa.

Nếu khách hàng không lấy được vé nào, hãy in -1 .

Ràng buộc

$$1 \leq n, m \leq 2 \cdot 10^5$$

$$1 \leq t_i, h_i \leq 10^9$$

Ví dụ

Input	Output
5 3	3
5 3 7 8 5	8
4 8 3	-1

Link submit : <https://cses.fi/problemset/task/1091/>

Code tham khảo : <https://ideone.com/lc5Psa>

Bài 56. Restaurant Customer

Bạn được cho biết thời gian đến và đi của n khách hàng trong một nhà hàng.

Số lượng khách hàng có mặt tại cửa hàng ở 1 thời điểm nhiều nhất là bao nhiêu?

Input

Dòng nhập đầu tiên có số nguyên n : số lượng khách hàng.

Sau đó, có n dòng mô tả khách hàng. Mỗi dòng có hai số nguyên a và b : thời gian đến và đi của một khách hàng.

Bạn có thể cho rằng tất cả thời gian đến và đi là khác nhau.

Output

In một số nguyên: số lượng khách hàng tối đa.

Ràng buộc

$$1 \leq n, m \leq 2 \cdot 10^5$$

$$1 \leq a, b \leq 10^9$$

Ví dụ

Input	Output
3 5 8 2 4 3 9	2
4 1 10 2 4 3 5 7 9	3

Giải thích test 1 : người khách (2,4) và (3,9) cùng có mặt tại cửa hàng, hoặc người khách (3,9) và (5,8) cùng có mặt tại cửa hàng

Giải thích test 2 : 3 người khách (1, 10), (2,4) và (3,5) cùng có mặt tại cửa hàng, ví dụ tại thời điểm $t = 3$.

Link submit : <https://cses.fi/problemset/task/1619/>

Code tham khảo : <https://ideone.com/kYyGoK>

Bài 57. Liên hoan phim 1

Trong một liên hoan phim, n bộ phim sẽ được chiếu. Bạn biết thời gian bắt đầu và kết thúc của mỗi bộ phim. Số lượng phim tối đa bạn có thể xem toàn bộ là bao nhiêu? Biết rằng nếu thời gian kết thúc của bộ phim trước bằng hoặc nhỏ hơn thời gian bắt đầu của bộ phim sau thì bạn có thể xem cả 2 phim này.

Input

Dòng nhập đầu tiên có số nguyên n : số lượng phim.

Sau đó, có n dòng mô tả các bộ phim. Mỗi dòng có hai số nguyên a và b : thời gian bắt đầu và kết thúc của một bộ phim.

Output

In một số nguyên: số lượng phim tối đa.

Ràng buộc

$$1 \leq n, m \leq 2 \cdot 10^5$$

$$1 \leq a, b \leq 10^9$$

Ví dụ

Input	Output
3 3 5 4 9 5 8	2

Giải thích test : Bạn có thể xem 2 bộ phim (3,5) và (5,8)

Link Submit : <https://cses.fi/problemset/task/1619/>

Code tham khảo : <https://ideone.com/7O9vHF>

Bài 58. Tìm 2 số có tổng bằng x

Bạn được cung cấp một mảng gồm n số nguyên và nhiệm vụ của bạn là tìm hai giá trị (ở các vị trí khác nhau) có tổng là x.

Input

Dòng đầu tiên có hai số nguyên n và x: kích thước mảng và tổng mục tiêu.

Dòng thứ hai có n số nguyên a1, a2,..., an: các giá trị của mảng.

Output

In ra hai số nguyên: vị trí của các giá trị. Nếu có một số giải pháp, bạn có thể in bất kỳ giải pháp nào trong số chúng. Nếu không có giải pháp nào, hãy in "IMPOSSIBLE".

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq a_i, x \leq 10^9$$

Ví dụ

Input	Output
4 8 2 7 5 1	2 4
1 2 1	IMPOSSIBLE

Link submit : <https://cses.fi/problemset/task/1640>

Code tham khảo : <https://ideone.com/a7o6YR>

Bài 59. Tìm 3 số có tổng bằng x

Bạn được cung cấp một mảng gồm n số nguyên và nhiệm vụ của bạn là tìm ba giá trị (ở các vị trí khác nhau) có tổng là x.

Input

Dòng đầu tiên có hai số nguyên n và x: kích thước mảng và tổng mục tiêu.

Dòng thứ hai có n số nguyên a_1, a_2, \dots, a_n : các giá trị của mảng.

Output

In ra hai số nguyên: vị trí của các giá trị. Nếu có một số giải pháp, bạn có thể in bất kỳ giải pháp nào trong số chúng. Nếu không có giải pháp nào, hãy in “IMPOSSIBLE”.

Ràng buộc

$$1 \leq n \leq 5000$$

$$1 \leq a_i, x \leq 10^9$$

Ví dụ

Input	Output
4 8 2 7 5 1	1 3 4

Link Submit : <https://cses.fi/problemset/task/1641>

Code tham khảo : <https://ideone.com/VRHtGB>

Bài 60. Tìm 4 số có tổng bằng x

Bạn được cung cấp một mảng gồm n số nguyên và nhiệm vụ của bạn là tìm bốn giá trị (ở các vị trí khác nhau) có tổng là x.

Input

Dòng đầu tiên có hai số nguyên n và x: kích thước mảng và tổng mục tiêu.

Dòng thứ hai có n số nguyên a_1, a_2, \dots, a_n : các giá trị của mảng.

Output

In ra hai số nguyên: vị trí của các giá trị. Nếu có một số giải pháp, bạn có thể in bất kỳ giải pháp nào trong số chúng. Nếu không có giải pháp nào, hãy in “IMPOSSIBLE”.

Ràng buộc

$$1 \leq n \leq 1000$$

$$1 \leq a_i, x \leq 10^9$$

Ví dụ

Input	Output
8 15 3 2 5 8 1 3 2 3	2 4 6 7

Link submit : <https://cses.fi/problemset/task/1642/>

Code tham khảo : <https://ideone.com/Nn0DMN>

Bài 61. Missing Coin Sum

Bạn có n đồng xu với các giá trị nguyên dương. Tổng nhỏ nhất mà bạn không thể tạo bằng cách sử dụng một tập hợp con của các đồng xu là bao nhiêu?

Input

Dòng nhập đầu tiên có số nguyên n : số xu.

Dòng thứ hai có n số nguyên x_1, x_2, \dots, x_n : giá trị của mỗi đồng xu.

Output

In một số nguyên: tổng xu nhỏ nhất.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq x_i \leq 10^9$$

Ví dụ

Input	Output
5 2 9 1 2 7	6
10 1 2 3 4 8 9 20 29 5 1	83

Link submit : <https://cses.fi/problemset/task/2183/>

Code tham khảo : <https://ideone.com/4XQtKV>

Bài 62. Collecting Number

Bạn được cung cấp một mảng chứa mỗi số từ $1 \dots n$ đúng một lần. Nhiệm vụ của bạn là thu thập các số từ 1 đến n theo thứ tự tăng dần.

Trên mỗi vòng, bạn đi qua mảng từ trái sang phải và thu thập càng nhiều số càng tốt.
Tổng số vòng sẽ là bao nhiêu?

Input

Dòng đầu tiên có số nguyên n : kích thước mảng.

Dòng tiếp theo có n số nguyên x_1, x_2, \dots, x_n : các số trong mảng.

Output

In một số nguyên: số vòng.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

Ví dụ

Input	Output
5 4 2 1 5 3	3
8 2 1 8 5 4 7 6 3	6

Giải thích test : Vòng 1 chọn số 1, vòng 2 chọn số 2 và 3, vòng 3 chọn số 4 và 5

Ở mỗi vòng bạn được chọn số x nếu như tất cả các số từ 1 tới $x-1$ đã được chọn trước đó rồi, mỗi vòng bạn có thể chọn nhiều số cùng 1 lúc.

Link submit : <https://cses.fi/problemset/task/2216/>

Code tham khảo : <https://ideone.com/YEZL2f>

Bài 63. Đếm mảng con có tổng bằng x

Cho một mảng gồm n số nguyên dương, nhiệm vụ của bạn là đếm số mảng con (dãy con các phần tử liên tiếp) có tổng bằng x .

Input

Dòng đầu tiên có hai số nguyên n và x : kích thước của mảng và tổng mục tiêu x .

Dòng tiếp theo có n số nguyên a_1, a_2, \dots, a_n : các phần tử trong mảng

Output

In một số nguyên: số lượng mảng con cần thiết.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq x, a_i \leq 10^9$$

Ví dụ

Input	Output
5 7 2 4 1 2 7	3

Link submit : <https://cses.fi/problemset/task/1660/>

Code tham khảo : <https://ideone.com/cVj1Jp>

Bài 64. Đếm mảng con có tổng bằng x(2)

Cho một mảng gồm n số nguyên dương, nhiệm vụ của bạn là đếm số mảng con (dãy con các phần tử liên tiếp) có tổng bằng x.

Input

Dòng đầu tiên có hai số nguyên n và x: kích thước của mảng và tổng mục tiêu x.

Dòng tiếp theo có n số nguyên a1, a2,..., an: các phần tử trong mảng

Output

In một số nguyên: số lượng mảng con cần thiết.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$-10^9 \leq x, a_i \leq 10^9$$

Ví dụ

Input	Output
5 7 2 4 1 2 7	3

Link submit : <https://cses.fi/problemset/task/1661>

Code tham khảo : <https://ideone.com/oiaMwW>

Bài 65. Đếm mảng con chia hết cho K.

Cho một mảng gồm n số nguyên, nhiệm vụ của bạn là đếm số mảng con (dãy con các phần tử liên tiếp) mà tổng các giá trị chia hết cho n.

Input

Dòng nhập đầu tiên có số nguyên n : kích thước của mảng.

Dòng tiếp theo có n số nguyên a_1, a_2, \dots, a_n : nội dung của mảng.

Output

In một số nguyên: số lượng mảng con thỏa mãn

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq a_i \leq 10^9$$

Ví dụ

Input	Output
100 2 1 -3 2 -7 7 -2 6 9 -4 10 -6 3 9 -8 7 -2 -9 4 -3 -2 6 6 3 7 2 -1 10 6 -4 4 9 -1 -5 -6 -9 1 2 2 -10 -2 3 3 4 3 - 6 -5 -1 9 6 -4 6 2 -1 6 1 6 1 3 7 -6 10 1 1 6 -9 0 5 -1 8 6 0 5 5 -3 1 1 -5 -9 -8 -9 -7 7 -6 10 7 8 1 -2 2 8 9 - 1 5 -7 3 -3 -9 -3 4	47

Link submit : <https://cses.fi/problemset/task/1662/>

Code tham khảo : <https://ideone.com/46Fp13>

Bài 66. Đếm mảng con có nhiều nhất k số khác nhau

Cho một mảng n số nguyên, nhiệm vụ của bạn là tính số mảng con liên tiếp có nhiều nhất k giá trị khác nhau.

Input

Dòng nhập đầu tiên có hai số nguyên n và k .

Dòng tiếp theo có n số nguyên x_1, x_2, \dots, x_n : nội dung của mảng.

Output

In một số nguyên: số mảng con.

Ví dụ

Input	Output
100 3	641

3 2 3 4 3 3 4 2 3 1 4 4 1 3 4 4 3 1 3 1 4 2 2 3 4 3 2 1 1 1 4 1 1 2 2 1 3 2 4 3 1 3 4 2 1 3 2 2 2 1 4 4 1 4 3 3 3 1 2 1 2 3 1 2 4 3 1 2 4 3 1 4 3 2 1 4 3 4 1 2 3 3 2 2 2 4 4 4 3 2 2 3 4 2 4 2 4 3 1 1	
5 2 1 2 3 1 1	10

Gợi ý : Dùng sliding window

Link submit : <https://cses.fi/problemset/task/2428/>

Code tham khảo : <https://ideone.com/BPxeBS>

Bài 67. Unique Subarray : Mảng con dài nhất mà mỗi phần tử chỉ xuất hiện 1 lần

Bạn được cung cấp một danh sách phát của một đài phát thanh kể từ khi đài đó được thành lập. Danh sách bài hát có tổng cộng n bài hát.

Chuỗi các bài hát liên tiếp dài nhất mà mỗi bài hát là duy nhất?

Input

Dòng đầu tiên chứa một số nguyên n: số lượng bài hát.

Dòng tiếp theo có n số nguyên k1, k2,..., kn: số id của mỗi bài hát.

Output

In độ dài của chuỗi bài hát dài nhất mà các bài hát này mỗi bài hát chỉ xuất hiện 1 lần.

Ví dụ

Input	Output
5 1 2 3 4 5	5
5 1 1 1 1 1	1
8 1 2 1 3 2 7 4 2	5

Link Submit : <https://cses.fi/problemset/task/1141/>

Code tham khảo : <https://ideone.com/OP9jzY>

Bài 68. Chia mảng thành k mảng con liên tiếp có tổng lớn nhỏ nhất

Bạn được cung cấp một mảng chứa n số nguyên dương.

Nhiệm vụ của bạn là chia mảng thành k mảng con liên tiếp sao cho tổng lớn nhất trong một mảng con càng nhỏ càng tốt.

Input

Dòng đầu tiên chứa hai số nguyên n và k: kích thước của mảng và số mảng con trong phép chia.

Dòng tiếp theo chứa n số nguyên x_1, x_2, \dots, x_n : nội dung của mảng.

Output

In một số nguyên: tổng lớn nhất trong một mảng con trong phép chia tối ưu.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq k \leq n$$

$$1 \leq x_i \leq 10^9$$

Ví dụ

Input	Output
5 3 2 4 7 3 5	8

Link submit : <https://cses.fi/problemset/task/1085/>

Code tham khảo : <https://ideone.com/epSaPy>

Bài 69.Sliding Median : Trung vị của cửa sổ

Bạn được cung cấp một mảng n số nguyên. Nhiệm vụ của bạn là tính giá trị trung bình của mỗi cửa sổ gồm k phần tử, từ trái sang phải.

Trung vị là phần tử ở giữa khi các phần tử được sắp xếp. Nếu số phần tử là chẵn, có thể có hai trung bình và chúng tôi giả định rằng trung vị là nhỏ hơn trong số đó.

Input

Dòng đầu tiên chứa hai số nguyên n và k: số phần tử và kích thước của cửa sổ.

Khi đó có n số nguyên x_1, x_2, \dots, x_n : nội dung của mảng.

Output

In ra n - k + 1 giá trị: trung điểm.

Ví dụ

Input	Output
8 3 2 4 3 5 8 1 2 1	3 4 5 5 2 1

Link submit : <https://cses.fi/problemset/task/1076>

Code tham khảo : <https://ideone.com/hwgBGT>

Bài 70. Kiểm tra đoạn lồng nhau

Phạm vi ở đây là một đoạn trên trục Ox với tọa độ bắt đầu và tọa độ kết thúc.

Cho n phạm vi, nhiệm vụ của bạn là xác định cho mỗi phạm vi xem nó có chứa một số phạm vi khác hay không và nếu một số phạm vi khác có chứa nó hay không.

Phạm vi $[a, b]$ chứa phạm vi $[c, d]$ nếu $a \leq c$ và $d \leq b$.

Input

Dòng nhập đầu tiên có số nguyên n: số dãy.

Sau đó, có n dòng mô tả các phạm vi. Mỗi dòng có hai số nguyên x và y: phạm vi là $[x, y]$.

Bạn có thể cho rằng không có phạm vi nào xuất hiện nhiều hơn một lần trong đầu vào.

Output

Đầu tiên hãy in một dòng mô tả cho từng phạm vi (theo thứ tự đầu vào) nếu nó có chứa một số phạm vi khác (1) hoặc không (0).

Sau đó, in một dòng mô tả cho từng phạm vi (theo thứ tự đầu vào) nếu một số phạm vi khác có chứa nó (1) hoặc không (0).

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq x < y \leq 10^9$$

Ví dụ

Input	Output
4 1 6 2 4 4 8 3 6	1 0 0 0 0 1 0 1

Link submit : <https://cses.fi/problemset/task/2168/>

Code tham khảo : <https://ideone.com/Ryg3DS>

Bài 71. Phân phối phòng

Có một khách sạn lớn, và n khách hàng sẽ đến sớm. Mỗi khách hàng muốn có một phòng duy nhất.

Bạn biết ngày đến và đi của từng khách hàng. Hai khách hàng có thể ở cùng một phòng nếu ngày đi của khách hàng thứ nhất sớm hơn ngày đến của khách hàng thứ hai.

Số lượng phòng tối thiểu cần thiết để chứa tất cả khách hàng là bao nhiêu? Và các phòng có thể được phân bổ như thế nào?

Input

Dòng đầu tiên chứa số nguyên n : số lượng khách hàng.

Sau đó, có n dòng, mỗi dòng mô tả một khách hàng. Mỗi dòng ghi hai số nguyên a và b : ngày đến và ngày đi.

Output

In đầu tiên một số nguyên k : số lượng phòng tối thiểu cần thiết.

Sau đó, in ra một dòng chứa số phòng của từng khách hàng theo thứ tự như trong đầu vào. Các phòng được đánh số $1, 2, \dots, k$. Bạn có thể in bất kỳ giải pháp hợp lệ nào.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq a \leq b \leq 10^9$$

Ví dụ

Input	Output
3 1 2 2 4 4 4	1 2 1

Link submit : <https://cses.fi/problemset/task/1164/>

Code tham khảo : <https://ideone.com/grudca>

Bài 72. Factory Machine

Một nhà máy có n máy có thể được sử dụng để tạo ra sản phẩm. Mục tiêu của bạn là tạo ra tổng số sản phẩm.

Đối với mỗi máy, bạn biết số giây nó cần để tạo ra một sản phẩm. Các máy có thể hoạt động đồng thời và bạn có thể tự do quyết định lịch trình của chúng.

Thời gian ngắn nhất cần thiết để làm ra t sản phẩm là bao nhiêu?

Input

Dòng nhập đầu tiên có hai số nguyên n và t : số lượng máy móc và sản phẩm.

Dòng tiếp theo ghi n số nguyên k_1, k_2, \dots, k_n : thời gian cần thiết để tạo ra một sản phẩm sử dụng mỗi máy.

Output

In một số nguyên: thời gian tối thiểu cần thiết để tạo ra t sản phẩm.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq t \leq 10^9$$

$$1 \leq k_i \leq 10^9$$

Ví dụ

Input	Output
3 7 3 2 5	8

Link submit : <https://cses.fi/problemset/task/1620/>

Code tham khảo : <https://ideone.com/IEFfcg>

Bài 73. Task and Deadline

Bạn phải xử lý n nhiệm vụ. Mỗi nhiệm vụ đều có thời gian cần để hoàn thành và thời hạn, và bạn sẽ xử lý các nhiệm vụ theo thứ tự lần lượt. Phần thưởng của bạn cho một nhiệm vụ là $d - f$ trong đó d là deadline của nó và f là thời gian hoàn thành nhiệm vụ. (Thời gian bắt đầu là 0 và bạn phải xử lý tất cả các nhiệm vụ ngay cả khi một nhiệm vụ sẽ mang lại phần thưởng âm.)

Phần thưởng tối đa của bạn là bao nhiêu nếu bạn thực hiện các hành động theo thứ tự tối ưu?

Input

Dòng nhập đầu tiên có số nguyên n : số nhiệm vụ.

Sau đó, có n dòng mô tả các nhiệm vụ. Mỗi dòng có hai số nguyên a và d : thời hạn và thời hạn của nhiệm vụ.

Output

In một số nguyên: phần thưởng tối đa.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq a, d \leq 10^6$$

Ví dụ

Input	Output
3 6 10 8 15 5 12	2

Link submit : <https://cses.fi/problemset/task/1630/>

Code tham khảo : <https://ideone.com/hmcSBI>

Bài 74. Đếm đoạn lồng nhau

Cho n đoạn, nhiệm vụ của bạn là đếm cho mỗi đoạn có bao nhiêu đoạn khác mà nó chứa và bao nhiêu đoạn khác chứa nó.

Đoạn $[a, b]$ chứa đoạn $[c, d]$ nếu $a \leq c$ và $d \leq b$.

Input

Dòng nhập đầu tiên có số nguyên n : số dãy.

Sau đó, có n dòng mô tả các đoạn. Mỗi dòng có hai số nguyên x và y : đoạn là $[x, y]$.

Bạn có thể cho rằng không có đoạn nào xuất hiện nhiều hơn một lần trong đầu vào.

Output

Đầu tiên hãy in một dòng mô tả cho mỗi dải ô (theo thứ tự đầu vào) có bao nhiêu đoạn khác mà nó chứa.

Sau đó, in một dòng mô tả cho mỗi phạm vi (theo thứ tự đầu vào) có bao nhiêu đoạn khác chứa nó.

Ràng buộc

$$1 \leq n \leq 2 \cdot 10^5$$

$$1 \leq x < y \leq 10^9$$

Ví dụ

Input	Output
4	2 0 0 0
1 6	0 1 0 1
2 4	
4 8	
3 6	

Gợi ý: Sử dụng Fenwick Tree thay vì prefix sum array

Link submit : <https://cses.fi/problemset/task/2169>

Code tham khảo : <https://cses.fi/paste/ab15a9545c0ffe8e2825be/>

Bài 75. Liên hoan phim 2

Trong một liên hoan phim, n bộ phim sẽ được trình chiếu. Câu lạc bộ điện ảnh của Syrjälä bao gồm k thành viên, tất cả sẽ tham gia liên hoan.

Bạn biết thời gian bắt đầu và kết thúc của mỗi bộ phim. Tổng số phim tối đa mà các thành viên câu lạc bộ có thể xem hoàn toàn nếu họ xem phim một cách tối ưu là bao nhiêu?

Input

Dòng đầu tiên có hai số nguyên n và k : số lượng phim và thành viên câu lạc bộ.

Sau đó, có n dòng mô tả các bộ phim. Mỗi dòng ghi hai số nguyên a và b : thời gian bắt đầu và kết thúc của một bộ phim.

Output

In một số nguyên: tổng số phim tối đa.

Ràng buộc

$$1 \leq k \leq n \leq 2 \cdot 10^5$$

$$1 \leq a < b \leq 10^9$$

Ví dụ

Input	Output
5 2	4
1 5	
8 10	

3 6 2 5 6 9	
10 3 32 79 91 96 45 96 5 44 7 56 20 43 62 84 65 87 27 42 67 88	7

Link submit : <https://cses.fi/problemset/task/1632/>

Code tham khảo : <https://cses.fi/paste/ea80a134fa00a6a1283507/>

Bài 76. Sliding Window Maximum

Cho mảng số nguyên gồm n phần tử và số nguyên dương k . Hãy tìm số lớn nhất của từng cửa sổ có kích thước k ($n - k + 1$)

Input

Dòng đầu tiên là 2 số nguyên dương n và k

Dòng thứ 2 là các phần tử trong mảng

Output

In ra số lớn nhất của từng cửa sổ

Ví dụ

Input	Output
8 3 1 3 -1 -3 5 3 6 7	3 3 5 5 6 7

Gợi ý: Sử dụng multiset, deque, hoặc maximum queue.