

Bài 1.

Cho 2 chuỗi A, B độ dài không vượt quá 10^6 . Đưa ra những vị trí xuất hiện chuỗi A trong chuỗi B.

Dữ liệu: Vào từ file FOUNDSTRING.INP

- Dòng đầu chứa chuỗi A
- Dòng hai chứa chuỗi B

Kết quả: Đưa ra file FOUNDSTRING.OUT

- Dòng đầu chứa số k là số vị trí xuất hiện chuỗi A trong chuỗi B.
- Dòng 2 chứa k số nguyên tăng dần xác định k vị trí xuất hiện A trong B.

Ví dụ:

FOUNDSTRING.INP	FOUNDSTRING.OUT
viet	3
vietnamnamvietviet	1 11 15

Bài 2.

Cho hai số nguyên dương N và K.

Yêu cầu: hãy đếm xem có bao nhiêu bộ ba số nguyên dương (a, b, c) không lớn hơn N thỏa mãn điều kiện $a + b$, $b + c$ và $c + a$ là bội số của K.

Dữ liệu: Vào từ file văn bản **MULK.INP** gồm một dòng chứa hai số nguyên N và K ($1 \leq N, K \leq 2 \times 10^5$).

Kết quả: Ghi ra file văn bản **MULK.OUT** gồm một dòng duy nhất là kết quả bài toán.

Ví dụ:

MULK.INP	MULK.OUT	Giải thích
3 2	9	Có 9 bộ (a, b, c): (1, 1, 1), (1, 1, 3), (1, 3, 1), (1, 3, 3), (2, 2, 2), (3, 1, 1), (3, 1, 3), (3, 3, 1) và (3, 3, 3).

Bài 3.

Bờm nhận được một ví tiền và muốn trả lại người đánh mất. Bờm bèn đăng thông tin về chiếc ví lên mạng xã hội với hy vọng thông tin đó sẽ được chủ nhân của chiếc ví đọc được. Mạng xã hội DailyShare mà Bờm tham gia có cơ chế hoạt động rất đặc biệt: Khi một người chia sẻ một thông tin lên mạng, một ngày sau, hệ thống kiểm duyệt mới cho phép bài được đăng lên cho tất cả những người theo dõi (follow) anh ta đọc được. Giả sử rằng tất cả những người này sẽ chia sẻ lại bài viết (sharing) ngay lập tức và cũng tương tự như vậy, những bài chia sẻ lại này cũng mất một ngày sau mới được đăng lên... Giả sử mạng xã hội đang có n người đánh số từ 0 tới $n - 1$. Bờm là người số 0 và người chủ chiếc ví là người số hiệu $n - 1$. Người ta biết được m thông tin, mỗi thông tin cho bởi cặp số nguyên (u, v) cho biết người u được theo dõi (followed) bởi người v trên DailyShare. Câu hỏi đặt ra là sau bao nhiêu ngày thì người chủ chiếc ví sẽ biết được thông tin.

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

- Dòng 1 chứa số hai số nguyên n, m ($2 \leq n \leq 10^5$; $1 \leq m \leq 10^5$)
- m dòng tiếp theo, mỗi dòng chứa hai số nguyên u, v cho biết người u được theo dõi bởi người v.

Kết quả: Ghi ra file văn bản **BROADCAST.OUT** một số nguyên duy nhất là số ngày phải chờ đợi mà người chủ chiếc ví biết được thông tin. Nếu không có cách nào để thông tin đến được chủ nhân của chiếc ví, ghi ra -1

Ví dụ:

BROADCAST.INP	BROADCAST.OUT
4 6 0 1 0 2 1 3 2 3 1 2 2 1	2

Bài 4.

Cho dãy số nguyên $A = (a_1, a_2, \dots, a_n)$ và một số nguyên dương $k \leq n$. Với mỗi giá trị i ($1 \leq i \leq n - k + 1$), hãy xác định giá trị nhỏ nhất trong k phần tử liên tiếp: $a_i, a_{i+1}, \dots, a_{i+k-1}$

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

- Dòng 1 chứa hai số nguyên dương $n \leq 5 \cdot 10^5$, $k \leq n$ cách nhau bởi dấu cách
- Dòng 2 chứa n số nguyên dương a_1, a_2, \dots, a_n ($\forall i: a_i \leq 10^6$)

Kết quả: Ghi ra file văn bản **MINIMUM.OUT** $n - k + 1$ dòng, dòng thứ i ghi giá trị nhỏ nhất trong các phần tử $a_i, a_{i+1}, \dots, a_{i+k-1}$

Ví dụ:

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

Bài 5.

Cho dãy số nguyên $A = (a_0, a_1, \dots, a_{n-1})$ hãy tìm một dãy con dài nhất gồm các phần tử liên tiếp trong A sao cho độ chênh lệch giữa phần tử lớn nhất và phần tử nhỏ nhất của dãy con đó không vượt quá Δ .

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

- Dòng 1 chứa số nguyên dương $n \leq 10^6$ và số nguyên không âm $\Delta \leq 2 \cdot 10^9$
- Dòng 2 chứa n số nguyên a_0, a_1, \dots, a_{n-1} cách nhau bởi dấu cách ($\forall i: |a_i| \leq 10^9$)

Kết quả: Ghi ra file văn bản **LMINMAX.OUT** một số nguyên duy nhất là độ dài dãy con tìm được.

Ví dụ:

LMINMAX.INP	LMINMAX.OUT
10 6 10 5 4 3 2 1 9 8 7 6	5

Bài 6.

Giáo sư X dự định thực hiện một chuyến đi bằng ô tô trên con đường dài n km tính từ km 0 (nơi xuất phát) tới km n (nơi kết thúc). Ô tô của giáo sư X có bình xăng dung tích là k lít, mỗi lít xăng cho phép ô tô đi được quãng đường dài đúng 1 km. Tại mỗi mốc km, từ mốc km 0 tới mốc km $n - 1$, có một trạm xăng, tại đó giáo sư X có thể mua thêm xăng nạp vào bình, tuy nhiên bình xăng không thể chứa quá k lít tính cả lượng xăng còn lại trong xe trước khi mua. Giá xăng ở trạm xăng

tại mốc km thứ i là c_i một lít ($\forall i: 0 \leq i < n$). Hãy tìm cách thực hiện chuyến đi với tổng số tiền mua xăng thấp nhất. Biết rằng giáo sư X xuất phát từ km số 0 với một bình xăng rỗng.

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

- Dòng 1 chứa hai số nguyên dương n, k ($k \leq n \leq 10^6$)
- Dòng 2 chứa n số nguyên dương c_0, c_1, \dots, c_{n-1} ($\forall i: c_i \leq 10^9$)

Kết quả: Ghi ra file văn bản **GAS.OUT** một số nguyên duy nhất là tổng số tiền mua xăng theo phương án tìm được.

Ví dụ

GAS.INP	GAS.OUT	Giải thích
9 3 1 7 2 9 3 6 8 5 4	22	

Bài 7.

Một cơ sở sửa chữa ô tô có nhận n chiếc xe để sửa, các xe được đánh số từ 0 tới $n - 1$. Do các nhân viên làm việc quá lười nhác nên đã đến hạn trả cho khách hàng mà vẫn chưa tiến hành sửa được chiếc xe nào. Theo hợp đồng đã ký kết từ trước, nếu bàn giao xe thứ i quá hạn ngày nào thì sẽ phải trả thêm một khoản tiền phạt là a_i .

Ông chủ cơ sở sửa chữa quyết định sa thải toàn bộ công nhân và thuê nhân công mới. Với lực lượng mới này, ông ta dự định rằng để sửa chiếc xe thứ i sẽ cần b_i ngày. Vấn đề đặt ra đối với ông là phải lập lịch sửa tuần tự các chiếc xe sao cho tổng số tiền bị phạt là ít nhất.

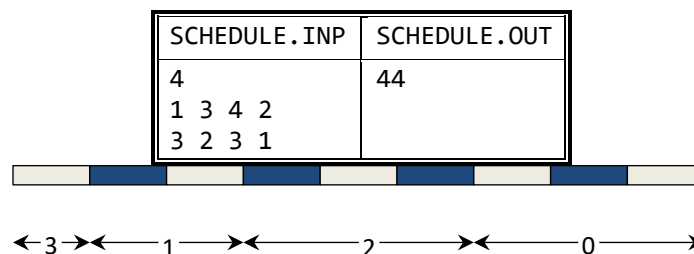
Yêu cầu: Hãy lập lịch sửa xe giúp cho ông chủ cơ sở sửa chữa ô tô sao cho tổng số tiền bị phạt là ít nhất.

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

- Dòng 1: Chứa số nguyên dương $n \leq 10^5$
- Dòng 2: Chứa n số nguyên dương $a_0, a_1, \dots, a_{n-1}, \forall i: a_i \leq 1000$
- Dòng 3: Chứa n số nguyên dương $b_0, b_1, \dots, b_{n-1}, \forall i: b_i \leq 1000$

Kết quả: Ghi ra file văn bản **SCHEDULE.OUT** một số nguyên duy nhất là số tiền phạt tối thiểu theo phương án tìm được.

Ví dụ



Phương án tối ưu là sửa các xe theo thứ tự 3, 1, 2, 0. Tiền phạt:

Xe 3: Muộn 1 (ngày) $\times 2 = 2$

Xe 1: Muộn 3 (ngày) $\times 3 = 9$

Xe 2: Muộn 6 (ngày) $\times 4 = 24$

Xe 0: Muộn 9 (ngày) $\times 1 = 9$

 Tổng cộng = 44

Nếu sửa theo thứ tự 0, 1, 2, 3 thì:

Xe 0: Muộn 3 (ngày) \times 1 = 3

Xe 1: Muộn 5 (ngày) \times 3 = 15

Xe 2: Muộn 8 (ngày) \times 4 = 32

Xe 3: Muộn 9 (ngày) \times 2 = 18

Tổng cộng = 68

Bài 8.

Siêu thị BC hiện đang có một chương trình khuyến mại dành riêng cho các khách hàng trung thành. Bằng việc mua các sản phẩm độc quyền của siêu thị, khách hàng có thể thu thập m loại phiếu giải thưởng, mỗi loại phiếu được đánh số từ 1 đến m.

Các phiếu giải thưởng này được sử dụng trong n chương trình giảm giá của siêu thị. Cụ thể, để có thể nhận được ưu đãi giảm giá của chương trình thứ i, khách hàng cần nộp lại cho siêu thị k_i phiếu thuộc các loại $T_{i1}, T_{i2}, \dots, T_{i_{k_i}}$ và sau đó khách hàng đó sẽ được giảm giá x_i đơn vị tiền mặt. Mỗi phiếu chỉ có thể được sử dụng cho một lần tham dự một chương trình giảm giá, tuy nhiên bạn có thể tham dự một chương trình giảm giá nhiều lần. Không có hai chương trình giảm giá nào yêu cầu cùng một loại phiếu. Một số loại phiếu là phiếu bẫy siêu thị và không thể được sử dụng ở bất cứ chương trình giảm giá nào.

Yêu cầu: Dũng đã sưu tập được rất nhiều phiếu giải thưởng của siêu thị BC. Hãy giúp Dũng tính xem nếu anh tham dự các chương trình giảm giá một cách hợp lí, anh có thể tiết kiệm được tối đa bao nhiêu đơn vị tiền mặt?

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

Dòng đầu tiên gồm một số nguyên t ($0 \leq t \leq 1000$) là số lượng bộ dữ liệu. Các bộ dữ liệu sẽ xuất hiện lần lượt theo định dạng:

- Dòng đầu tiên gồm một số nguyên dương n và m ($1 \leq n \leq 30; 1 \leq m \leq 30$) lần lượt là số chương trình giảm giá và số phiếu.
- n dòng tiếp theo, mỗi dòng mô tả một chương trình giảm giá. Số đầu tiên của dòng thứ i là số loại phiếu cần dùng k_i ($1 \leq k_i \leq m$). k_i số nguyên tiếp theo, mỗi số là số thứ tự của một phiếu cần dùng (các số này đôi một khác nhau). Số nguyên x_i ($0 \leq x_i \leq 10^6$) cuối cùng của dòng là số đơn vị tiền mặt Dũng tiết kiệm được mỗi lần Dũng tham gia chương trình này.
- Dòng tiếp theo chứa m số nguyên, số thứ i là số phiếu loại i mà Dũng đang có. Dũng không có nhiều hơn 100 phiếu mỗi loại.

Kết quả: Ghi ra file văn bản **VOUCH.OUT** Gồm một dòng duy nhất chứa một số nguyên là số đơn vị tiền mặt Dũng có thể tiết kiệm được nếu tham gia các chương trình giảm giá một cách tối ưu nhất.

Ví dụ:

VOUCH.INP	VOUCH.OUT	Giải thích
1 3 6 2 1 2 100	2500	Dũng có thể tham gia chương trình khuyến mãi đầu tiên 1 lần, chương trình khuyến mãi thứ hai 3 lần và chương

3 3 4 5 200 1 6 300 1 2 3 4 5 6		trình khuyến mãi thứ ba 6 lần. Tổng số tiền Dũng có thể tiết kiệm là $100 \times 1 + 200 \times 3 + 300 \times 6 = 2500$
--	--	--