

## BÀI TẬP LẬP TRÌNH

### ĐOẠN CON LỚN NHẤT

Cho dãy gồm  $n$  số nguyên  $a_1, a_2, \dots, a_n$ . Hãy tìm tổng lớn nhất của một đoạn con gồm các phần tử liên tiếp của dãy.

**Dữ liệu:** Vào từ tập tin văn bản **MAXSUBARRAY.INP**

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 2 \times 10^5$ ) – số phần tử của dãy.
- Dòng tiếp theo chứa dãy gồm  $n$  số nguyên  $a_1, a_2, \dots, a_n$  ( $-10^9 \leq a_i \leq 10^9$ ).

**Kết quả:** Ghi ra tập tin văn bản **MAXSUBARRAY.OUT** một số nguyên là tổng lớn nhất tìm được.

**Ví dụ:**

MAXSUBARRAY . INP	MAXSUBARRAY . OUT
8 -1 3 -2 5 3 -5 2 2	9

## CÁC QUE TÍNH

Có  $n$  que tính, que tính thứ  $i$  có chiều dài  $l_i$ . Với mỗi que tính, ta có thể tăng hoặc giảm chiều dài của nó  $x$  đơn vị với cùng chi phí là  $x$ .

**Yêu cầu:** Hãy điều chỉnh độ dài các que tính sao cho chúng có cùng độ dài và có tổng chi phí là nhỏ nhất.

**Dữ liệu:** Vào từ tập tin văn bản **STICKSLEN.INP**

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 2 \times 10^5$ ) – số lượng que tính.
- Dòng tiếp theo chứa dãy gồm  $n$  số nguyên  $l_1, l_2, \dots, l_n$  ( $1 \leq l_i \leq 10^9$ ) – chiều dài của các que tính.

**Kết quả:** Ghi ra tập tin văn bản **STICKSLEN.OUT** một số nguyên là chi phí nhỏ nhất tìm được.

**Ví dụ:**

STICKSLEN . INP	STICKSLEN . OUT
5 2 3 1 5 2	5

## THỰC KHÁCH NHÀ HÀNG

Có  $n$  thực khách đến thưởng thức các món đặc sản của một nhà hàng. Với mỗi thực khách, ta biết thời điểm vào và thời điểm rời nhà hàng. Hãy cho biết số lượng nhiều nhất các thực khách có mặt trong nhà hàng tại một thời điểm nào đó.

**Dữ liệu:** Vào từ tập tin văn bản **RESTAURANT.INP**

- Dòng đầu tiên chứa số nguyên  $n (1 \leq n \leq 2 \times 10^5)$  – số lượng thực khách.
- Mỗi dòng trong  $n$  dòng tiếp theo chứa 2 số nguyên  $a, b (1 \leq a < b \leq 10^9)$  – thời điểm vào và thời điểm ra khỏi nhà hàng của thực khách tương ứng.

**Kết quả:** Ghi ra tập tin văn bản **RESTAURANT.OUT** một số nguyên là số thực khách nhiều có mặt trong nhà hàng ở một thời điểm nào đó.

**Ví dụ:**

RESTAURANT . INP	RESTAURANT . OUT
3 5 8 2 4 3 9	2

**VLTK**

Bờm là một học sinh rất mê game. Cậu tích cực ngày đêm cày cuốc để nhân vật của mình ngày càng mạnh lên. Gần đây do thầy giáo cho khá nhiều bài tập tin học nên cậu ta buộc phải giảm bớt cày cuốc game. Tuy nhiên cậu ta lại không muốn nhân vật của mình yếu đi và có thể bị tiêu diệt trong khi làm bài tập nên phải mua vật phẩm để tăng sức mạnh cho nhân vật.

Có 2 loại vật phẩm giúp tăng sức mạnh cho nhân vật: tăng trực tiếp và tăng theo tỉ lệ phần trăm. Cụ thể, nếu nhân vật có sức mạnh là  $b$  và có  $n$  vật phẩm tăng trực tiếp có giá trị tương ứng  $d_1, d_2, \dots, d_n$  và  $m$  vật phẩm tăng theo tỉ lệ phần trăm có giá trị tương ứng  $p_1, p_2, \dots, p_m$  thì sức mạnh của nhân vật sẽ tăng lên thành  $(b + d_1 + d_2 + \dots + d_n) * (100 + p_1 + p_2 + \dots + p_m) / 100$ .

Nhưng không may là game giới hạn số vật phẩm được phép sử dụng tối đa là  $k$  cho cả 2 loại và mỗi vật phẩm chỉ được phép sử dụng không quá 1 lần. HungDM đang đau đầu vì không biết nên chọn mua những vật phẩm nào để có lợi nhất.

**Yêu cầu:** Hãy giúp Bờm chọn vật phẩm để chỉ số sức mạnh của nhân vật được tăng tối đa.

**Dữ liệu:** Vào từ tập tin văn bản **VLTK.INP**

- Dòng đầu tiên chứa 3 số nguyên dương  $b, k, n, m$  ( $0 \leq b, k, n, m \leq 50000$ )
- Dòng thứ hai chứa dãy  $n$  số nguyên  $d_1, d_2, \dots, d_n$  ( $0 \leq d_i \leq 50000$ )
- Dòng thứ ba chứa dãy  $m$  số nguyên  $p_1, p_2, \dots, p_m$  ( $0 \leq p_i \leq 50000$ )

**Kết quả:** Ghi ra tập tin văn bản **VLTK.OUT**

- Dòng đầu tiên ghi 2 số  $nd, np$  ( $0 \leq nd \leq n; 0 \leq np \leq m; 0 \leq nd + np \leq k$ ) tương ứng với số vật phẩm tăng trực tiếp và tăng theo tỉ lệ phần trăm được chọn.
- Dòng thứ hai ghi chỉ số các vật phẩm loại tăng trực tiếp được chọn
- Dòng thứ ba ghi chỉ số các vật phẩm loại tăng theo tỉ lệ phần trăm được chọn

**Ví dụ:**

VLTK . INP	VLTK . OUT
70 3 2 2	2 1
40 30	2 1
50 40	1

## SẮP PHÒNG

Có  $n$  khách đăng ký thuê phòng khách sạn. Mỗi khách ở một phòng riêng và đăng ký với khách sạn ngày đến và ngày trả phòng. Hai khách có thể được bố trí phòng giống nhau nếu một khách có ngày trả phòng sớm hơn ngày đến của khách còn lại.

**Yêu cầu:** Hãy cho biết cách bố trí các phòng cho  $n$  khách thuê sao cho số lượng phòng sử dụng là ít nhất.

**Dữ liệu:** Vào từ tập tin văn bản **ROOMSALLOC.INP**

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 2 \times 10^5$ ) – số lượng khách đăng ký thuê phòng.
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo chứa 2 số nguyên  $a_i, b_i$  ( $1 \leq a_i \leq b_i \leq 10^9$ ) – ngày đến và ngày trả phòng của khách thứ  $i$ .

**Kết quả:** Ghi ra tập tin văn bản **ROOMSALLOC.OUT**

- Dòng đầu ghi số nguyên  $k$  – số phòng tối thiểu cần sử dụng. Các phòng dành cho thuê được đánh thứ tự từ 1 đến  $k$ .
- Dòng tiếp theo chứa dãy gồm  $n$  số nguyên, số thứ  $i$  là thứ tự phòng được bố trí cho khách thuê thứ  $i$ .

**Ví dụ:**

ROOMSALLOC . INP	ROOMSALLOC . OUT
3	2
1 2	1 2 1
2 4	
4 4	

## ĐỘ ĐO

Hai xâu ký tự được gọi là đảo của nhau nếu ta có thể hoán vị các ký tự của một xâu để được xâu còn lại. Ví dụ: xâu “occurs” là đảo của xâu “succor”, tuy nhiên xâu “dear” không phải là đảo của xâu “dared” (vì chữ ‘d’ xuất hiện 2 lần trong “dared” nhưng chỉ xuất hiện “dear” trong 1 lần).

Độ đo giữa hai xâu ký tự là số ký tự ít nhất cần phải xóa (trên cả hai xâu) để hai xâu còn lại đảo của nhau. Ví dụ độ đo giữa hai xâu “sleep” và “leap” là 3, độ đo giữa hai xâu “dog” và “cat” là 6.

**Yêu cầu:** Hãy tính độ đo giữa hai xâu cho trước.

**Dữ liệu:** Vào từ tập tin văn bản **AD.INP** gồm hai dòng, mỗi dòng chứa một xâu ký tự chỉ gồm các chữ cái tiếng Anh thường, mỗi dòng có không quá 1 triệu ký tự.

**Kết quả:** Ghi ra tập tin văn bản **AD.OUT** một số nguyên duy nhất là độ đo tính được.

**Ví dụ:**

AD . INP	AD . OUT
begin end	4

## XÂU CƠ SỞ

Lũy thừa nguyên bậc  $n$  của một xâu là việc lặp lại liên tiếp  $n$  lần xâu đó. Ví dụ  $(abc)^3 = abcabcabc$ . Xâu cơ sở của 2 xâu  $S$  và  $T$  là xâu  $q$  có độ dài lớn nhất sao cho tồn tại 2 số nguyên  $i$  và  $j$  để  $S = q^i$  và  $T = q^j$ .

**Yêu cầu:** Cho 2 xâu khác rỗng  $S$  và  $T$  có độ dài không quá  $10^6$  và chỉ chứa các ký tự latin thường. Hãy xác định xâu cơ sở của  $S$  và  $T$ . Nếu không tồn tại xâu cơ sở thì đưa ra thông báo **NO**.

**Dữ liệu:** Vào từ tập tin văn bản **BASESTRING.INP** gồm xâu  $S$  và xâu  $T$  trên 2 dòng khác nhau.

**Kết quả:** Ghi ra tập tin văn bản **BASESTRING.OUT** xâu  $q$  hoặc thông báo **NO**.

**Ví dụ:**

BASESTRING . INP	BASESTRING . OUT
aaa aa	a

## PYTHAGORE

Định lí Pythagore do Pythagoras, nhà toán học người Hy Lạp sống vào khoảng thế kỉ thứ 6 trước công nguyên, phát minh. Ngày nay, định lí Pythagore được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau như kiến trúc, xây dựng và đo lường. Những ứng dụng phổ biến của định lí Pythagore có thể kể ra là tính khoảng cách giữa 2 điểm trong mặt phẳng hoặc kiểm tra một tam giác có phải là tam giác vuông.

Định lí phát biểu như sau: Tam giác vuông có độ dài 3 cạnh  $a, b, c$ , trong đó  $c$  là độ dài cạnh huyền khi đó  $a^2 + b^2 = c^2$ .

Tí rất thích thú với tính chất này nhưng cậu ta chỉ quan tâm đến các tam giác vuông có độ dài ba cạnh  $a, b, c$  ( $0 < a < b < c$ ) là số nguyên. Tí nhận thấy rằng, với cùng một chu vi thì có thể tồn tại nhiều tam giác vuông khác nhau có cạnh nguyên. Chẳng hạn với chu vi 60, tồn tại 2 tam giác vuông là (10,24,26) và (15,20,25). Còn với chu vi 12 chỉ tồn tại 1 tam giác vuông (3,4,5).

**Yêu cầu:** Cho trước chu vi  $P$ . Đếm số lượng tam giác vuông khác nhau có độ dài ba cạnh nguyên  $a, b, c$  ( $a < b < c$ ) và chu vi là  $P$ .

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

- Dòng đầu tiên chứa số nguyên  $t$  ( $1 \leq t \leq 5000$ ) – số bộ test.
- Dòng thứ  $i$  trong  $t$  dòng tiếp theo chứa số nguyên  $P_i$  ( $1 \leq P_i \leq 5000$ ) – chu vi của các tam giác vuông thứ  $i$ .

**Kết quả:** Ghi ra file văn bản **PYTHAGORE.OUT** gồm  $t$  dòng, dòng thứ  $i$  ghi một số nguyên là số lượng tam giác vuông khác nhau có độ dài ba cạnh nguyên và chu vi  $P_i$ .

**Ví dụ:**

PYTHAGORE . INP	PYTHAGORE . OUT
2	1
12	2
60	



## MERLIN

Một hôm Merlin qua về tòa tháp của mình thì thấy tất cả  $n$  hũ rượu thuốc quý đều bị Morgana yểm bùa. Merlin biết cách gỡ bỏ bùa chú, nhưng điều này đòi hỏi các bình cần khử bùa phải có rượu và chứa một lượng rượu như nhau.

Sau một lúc suy nghĩ Merlin quyết định chọn một số bình, rót hết rượu từ những bình được chọn sang các bình còn lại sao cho chúng có cùng một lượng rượu. Những bình rỗng không thể gỡ bỏ bùa chú bị đập vỡ. Với những bình còn lại Merlin tiến hành xử lý gỡ bỏ bùa chú. Bản thân các bình đựng rượu đều rất đẹp và quý, vì vậy Merlin có gắng chọn cách làm sao cho số bình phải đập bỏ là ít nhất.

**Yêu cầu:** Hãy xác định số lượng bình tối thiểu phải đập bỏ.

**Dữ liệu:** Vào từ tập tin văn bản **MERLIN.INP**

- Dòng đầu chứa số nguyên  $n$  ( $2 \leq n \leq 10^5$ )
- Dòng thứ hai chứa dãy  $n$  số nguyên  $a_1, a_2, \dots, a_n$  – số lượng rượu trong các bình ( $1 \leq a_i \leq 10^9$ )

**Kết quả:** Ghi ra tập tin văn bản **MERLIN.OUT** số lượng bình tối thiểu bị đập bỏ.

**Ví dụ:**

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

## NỐI DÂY

Có  $n$  đoạn dây xanh,  $n$  đoạn dây đỏ,  $n$  đoạn dây tím và  $n$  đoạn dây vàng. Độ dài các đoạn dây được cho trước.

**Yêu cầu:** cho số nguyên  $L$ , hãy cho biết có bao nhiêu cách chọn đúng 1 đoạn dây xanh, 1 đoạn dây đỏ, 1 đoạn dây tím và 1 đoạn dây vàng để nối lại thành một sợi dây trang trí có độ dài bằng  $L$ . Hai cách chọn được gọi là khác nhau nếu có đoạn dây được chọn trong một cách nhưng không được chọn trong cách còn lại.

**Dữ liệu:** Vào từ tập tin văn bản **TERA.INP**

- Dòng 1 chứa hai số nguyên dương  $n \leq 1000$ ;  $L \leq 10^9$
- Dòng 2 chứa  $n$  số nguyên dương là độ dài  $n$  đoạn dây xanh
- Dòng 3 chứa  $n$  số nguyên dương là độ dài  $n$  đoạn dây đỏ
- Dòng 4 chứa  $n$  số nguyên dương là độ dài  $n$  đoạn dây tím
- Dòng 5 chứa  $n$  số nguyên dương là độ dài  $n$  đoạn dây vàng

Độ dài các đoạn dây không quá  $10^9$

**Kết quả:** Ghi ra tập tin văn bản **TERA.OUT** một số nguyên duy nhất là số cách chọn tính được.

**Ví dụ:**

TERA . INP	TERA . OUT
3 28 1 1 1 1 1 1 10 11 12 13 14 15	18

## ĐẠI DIỆN

Trên trục số cho  $n$  đoạn đóng, đoạn thứ  $i$  là  $[L_i, R_i]$ . Hãy chọn ra một tập ít nhất các điểm nguyên phân biệt trên trục số thoả mãn: Mỗi đoạn trong số  $n$  đoạn đã cho phải chứa tối thiểu 2 điểm trong tập này.

**Dữ liệu:** Vào từ tập tin văn bản **REP.INP**

- Dòng 1: chứa số nguyên dương  $n \leq 10^5$
- Trong  $n$  dòng tiếp theo, dòng thứ  $i$  chứa hai số nguyên  $L_i, R_i$  ( $\forall i: -10^6 \leq L_i < R_i \leq 10^6$ )

**Kết quả:** Ghi ra tập tin văn bản **REP.OUT**

- Dòng 1: Ghi số điểm được chọn  $m$
- Dòng 2: Ghi các toạ độ (trên trục số) của  $m$  điểm được chọn

**Ví dụ:**

REP . INP	REP . OUT
3 6 10 1 6 4 9	3 4 6 9

## VÒNG TRÒN SỐ

Trên một vòng tròn có đánh dấu  $n$  điểm phân biệt. Bắt đầu từ một điểm, theo chiều kim đồng hồ, người ta đánh số các điểm đã chọn từ 1 tới  $n$ , sau đó ghi vào vị trí mỗi điểm  $i$  một số nguyên  $a_i$  ( $\forall i = 1, 2, \dots, n$ )

**Yêu cầu:** Xác định vị trí hai điểm *phân biệt*  $p, q$  thỏa mãn: Nếu đi từ điểm  $p$  tới điểm  $q$  theo chiều kim đồng hồ trên vòng tròn thì dãy số nguyên đi qua có *thứ tự không giảm* và có *chiều dài (số phần tử) lớn nhất có thể*.

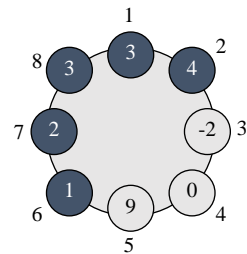
**Dữ liệu:** Vào từ tập tin văn bản **CLIS.INP**

- Dòng 1 chứa số nguyên dương  $n$ , ( $2 \leq n \leq 10^5$ )
- Dòng 2 chứa  $n$  số nguyên  $a_1, a_2, \dots, a_n$  ( $|a_i| \leq 10^9, \forall i = 1, 2, \dots, n$ ) theo đúng thứ tự đó.

**Kết quả:** Ghi ra tập tin văn bản **CLIS.OUT** hai vị trí  $p, q$  tìm được theo đúng thứ tự. Nếu có nhiều phương án cùng tối ưu, chỉ đưa ra một phương án.

**Ví dụ:**

CLIS.INP	CLIS.OUT
8 3 4 -2 0 9 1 2 3	6 2



## THÙNG CHỨA

Cho dãy gồm  $n$  số nguyên không âm  $a_1, a_2, \dots, a_n$ , giá trị  $a_i$  biểu diễn một điểm trong mặt phẳng có tọa độ  $(i, a_i)$ . Như vậy ta có  $n$  đường thẳng đứng nối từng cặp điểm  $(i, 0)$  và  $(i, a_i)$ .

**Yêu cầu:** Hãy tìm 2 đường thẳng khác nhau cùng với trục hoành tạo thành hình ảnh một thùng chứa sao cho có thể chứa được nhiều nước nhất.

**Dữ liệu:** Vào từ tập tin văn bản **CONTAINER.INP**

- Dòng đầu tiên chứa số nguyên  $n (2 \leq n \leq 10^6)$
- Dòng tiếp theo chứa dãy số nguyên  $a_1, a_2, \dots, a_n (0 \leq a_i \leq 10^9)$

**Kết quả:** Ghi ra tập tin văn bản **CONTAINER.OUT** một số nguyên là diện tích lớn nhất của mặt cắt thùng chứa.

**Ví dụ:**

CONTAINER . INP	CONTAINER . OUT
5 4 1 3 5 2	12

## DÃY PALINDROME

Xét dãy gồm  $n$  số nguyên  $a_1, a_2, \dots, a_n$ . Ta thực hiện phép biến đổi trên dãy như sau: chọn 2 phần tử nằm cạnh nhau và thay thế chúng bằng tổng của chúng. Như vậy sau mỗi phép biến đổi, số phần tử của dãy sẽ giảm đi 1.

**Yêu cầu:** Cho biết số phép biến đổi ít nhất cần phải thực hiện để dãy số tạo được sau các phép biến đổi là dãy đối xứng. Dãy có 1 phần tử cũng được gọi là dãy đối xứng.

**Dữ liệu:** Vào từ tập tin văn bản **PALINSEQ.INP**

- Dòng đầu tiên chứa số nguyên  $n (1 \leq n \leq 10^6)$ .
- Dòng tiếp theo chứa dãy  $a_1, a_2, \dots, a_n (1 \leq a_i \leq 10^9)$

**Kết quả:** Ghi ra tập tin văn bản **PALINSEQ.OUT** một số nguyên là số phép biến đổi ít nhất để biến dãy ban đầu thành dãy đối xứng.

**Ví dụ:**

PALINSEQ . INP	PALINSEQ . OUT
4 1 4 3 2	2

## SỐ NGUYÊN DƯƠNG NHỎ NHẤT

An, một người bạn của Giang đang tìm cách giải một bài toán liên quan đến số nguyên dương. Lúc này, An đang cần sự giúp đỡ của Giang. Thử thách lần này là một dãy gồm  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$  và tìm ra một số nguyên dương  $k$  nhỏ nhất không có trong dãy số đó.

Vì số lượng các số nguyên dương trong dãy số đã cho rất lớn nên việc tìm thủ công là không thể. Chính vì vậy, Giang cần một thuật toán để cài đặt vào máy tính và nhờ máy tính tìm giúp.

**Yêu cầu:** Hãy giúp Giang tìm số nguyên dương  $k$  nhỏ nhất không xuất hiện trong một dãy cho trước.

**Dữ liệu:** Vào từ tập tin văn bản **KMIN.INP** có cấu trúc như sau :

- Dòng đầu tiên là số nguyên dương  $n (1 \leq n \leq 10^7)$
- Trong  $n$  dòng sau, dòng thứ  $i$  là số nguyên dương  $a_i (i = 1, 2, \dots, n; a_i \leq 10^9)$

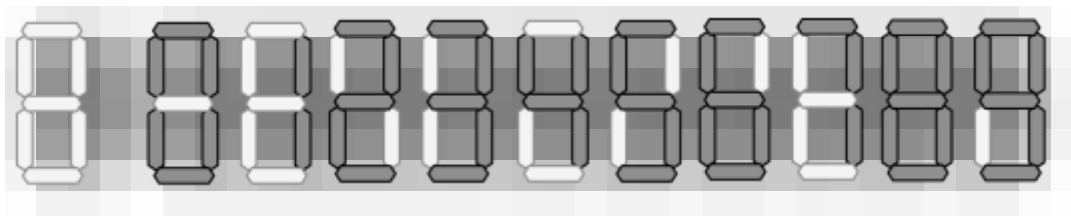
**Kết quả:** Ghi ra tập tin văn bản **KMIN.OUT** số nguyên dương  $k$  nhỏ nhất tìm được.

**Ví dụ :**

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

## HIỆN SỐ BẰNG ĐÈN LED

Quà sinh nhật mà Mai nhận được là một máy tính bấm tay màn hình tinh thể lỏng. Màn hình có thể hiển thị được  $n$  chữ số. Có 7 đèn LED tạo thành 7 vạch để hiển thị một chữ số. Mỗi chữ số sẽ tương ứng với một số đèn LED được kích hoạt và vạch tương ứng sẽ có màu đen. Cách hiển thị các số là như sau:



Như vậy, để hiển thị số 0 cần 6 vạch đen, số 1 cần 2 vạch đen, ... Là người ham hiểu biết, Mai tự hỏi không biết số gồm  $n$  chữ số có nghĩa nào nhỏ nhất và lớn nhất có thể hiển thị với đúng  $k$  vạch đen trên màn hình.

**Yêu cầu:** Cho  $n$  và  $k$  ( $1 \leq n \leq 100, 1 \leq k \leq 700$ ). Hãy xác định số gồm  $n$  chữ số có nghĩa nhỏ nhất và lớn nhất, mỗi số được hiển thị với đúng  $k$  vạch đen trên màn hình.

**Dữ liệu:** Vào từ tập tin văn bản **LED.INP** chứa 2 số nguyên  $n$  và  $k$

**Kết quả:** Ghi ra tập tin văn bản **LED.OUT**

- Dòng thứ nhất ghi số nguyên nhỏ nhất
- Dòng thứ hai ghi số nguyên lớn nhất
- Nếu không có nghiệm thì đưa ra thông báo NO SOLUTION

**Ví dụ:**

LED . INP	LED . OUT
5 15	10117 97111



## BỐ TRÍ PHÒNG HỌP

Có  $n$  cuộc họp đánh số từ 1 đến  $n$  đăng ký làm việc tại một phòng hội thảo. Cuộc họp  $i$  cần được bắt đầu ngay sau thời điểm  $s_i$  và kết thúc tại thời điểm  $f_i$ :  $(s_i, f_i]$ .

**Yêu cầu:** Cho biết có thể bố trí phòng hội thảo phục vụ được nhiều nhất bao nhiêu cuộc họp, sao cho khoảng thời gian làm việc của hai cuộc họp bất kỳ là không giao nhau.

**Dữ liệu:** Vào từ tập tin văn bản **MEETING.INP**

- Dòng đầu tiên chứa số nguyên dương  $n \leq 10^6$
- Dòng thứ  $i$  trong số  $n$  dòng tiếp theo chứa hai số nguyên  $s_i, f_i$  ( $0 \leq s_i < f_i \leq 10^5$ )

**Kết quả:** Ghi ra tập tin văn bản **MEETING.OUT**

- Dòng đầu tiên ghi số  $k$  là số các cuộc họp được chấp nhận phục vụ
- $k$  dòng tiếp theo liệt kê số hiệu các cuộc họp được chấp nhận theo thứ tự từ cuộc họp đầu tiên tới cuộc họp cuối cùng, mỗi dòng ghi số hiệu một cuộc họp.

**Ví dụ:**

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

## HAI TẬP

Cho tập gồm  $n$  số nguyên  $\{1, 2, \dots, n\}$ . Hãy chia tập thành 2 tập con có tổng bằng nhau.

**Dữ liệu:** Vào từ tập tin văn bản **TWOSETS.INP** chứa số nguyên  $n (1 \leq n \leq 10^6)$ .

**Kết quả:** Ghi ra tập tin văn bản **TWOSETS.OUT**

- Dòng đầu tiên ghi YES tương ứng với có cách chia, ngược lại in NO.
- Nếu có cách chia thì in ra một cách chia trên 4 dòng như sau:
  - + Dòng đầu tiên ghi số nguyên  $a$  – số phần tử của tập thứ nhất.
  - + Dòng thứ hai ghi  $a$  số nguyên là các phần tử của tập thứ nhất.
  - + Dòng thứ ba ghi số nguyên  $b$  – số phần tử của tập thứ hai.
  - + Dòng thứ tư ghi  $b$  số nguyên là các phần tử của tập thứ hai.

**Ví dụ:**

TWOSETS . INP	TWOSETS . OUT
7	YES 4 1 2 4 7 3 3 5 6
6	NO

## CHUỖI LẶP

Xét chuỗi ban đầu  $s$ . Gọi  $T(s, n)$  là chuỗi gồm  $n$  kí tự đầu tiên của chuỗi được tạo thành bằng cách viết lặp lại  $s$  vô hạn lần.

Chẳng hạn với  $s = \text{"abcac"}$  và với  $n = 10$  ta có  $T(s, n) = \text{"abcacabcac"}$ . Kí tự 'a' xuất hiện 4 lần trong chuỗi  $T$ .

**Yêu cầu:** Cho chuỗi  $s$  chỉ gồm các kí tự chữ cái latin in thường và số nguyên dương  $n$ . Hãy đếm tần số xuất hiện của kí tự 'a' trong chuỗi  $T(s, n)$ .

**Dữ liệu:** Vào từ tập tin văn bản **REPSTR.INP**

- Dòng đầu tiên chứa chuỗi  $s$  có độ dài không quá 100.
- Dòng thứ hai chứa số nguyên dương  $n (1 \leq n \leq 10^{12})$ .

**Kết quả:** Ghi ra tập tin văn bản **REPSTR.OUT** một số nguyên là kết quả của bài toán.

**Ví dụ:**

REPSTR . INP	REPSTR . OUT
aba 10	7
a 1000000	1000000

**CHUỖI ANAGRAM**

Hai chuỗi kí tự được gọi là anagram nếu ta có thể thay đổi thứ tự các kí tự trong chuỗi này để tạo thành chuỗi kia và ngược lại. Chẳng hạn các cặp chuỗi sau là anagram: ("taste", "state"), ("act", "cat"), ("listen", "slient"). Còn cặp chuỗi sau không phải là anagram: ("feel", "fell").

Xét chuỗi  $s$ , ta chia  $s$  thành 2 chuỗi con liên tiếp  $x$  và  $y$ , nghĩa là  $x = s_0s_1 \dots s_{i-1}$ ,  $y = s_is_{i+1} \dots s_{|s|-1}$  với  $i(0 < i < |s|)$  là vị trí tách. Nếu  $x, y$  không phải anagram ta sẽ thay thế một số kí tự để chúng trở thành anagram.

Chẳng hạn  $s = \text{"abccde"}$ , ta tách  $s$  thành  $x = \text{"abc"}$  và  $y = \text{"cde"}$ . Nếu thay kí tự 'a' bằng kí tự 'd', thay kí tự 'b' bằng kí tự 'e' trong chuỗi  $x$  thì  $x, y$  trở thành anagram với 2 phép thay thế.

**Yêu cầu:** Cho chuỗi  $s$ . Tìm số phép thay thế ít nhất để  $x, y$  trở thành anagram.

**Dữ liệu:** Vào từ tập tin văn bản **ANAGRAM.INP**

- Dòng đầu tiên chứa số nguyên  $t(1 \leq t \leq 100)$  – số bộ test.
- Mỗi dòng trong  $t$  dòng tiếp theo chuỗi  $s$  độ dài không quá 10000 các kí tự chữ cái latin in thường.

**Kết quả:** Ghi ra tập tin văn bản **ANAGRAM.OUT** gồm  $t$  dòng, mỗi dòng ghi số phép thay thế ít nhất cần thực hiện. Trường hợp không có lời giải thì ghi số  $-1$ .

**Ví dụ:**

ANAGRAM . INP	ANAGRAM . OUT
6	3
aaabbb	1
ab	-1
abc	2
mno	0
xyyx	1
xaxbbbx	

## TẠO CHUỖI ANAGRAM

Hai chuỗi kí tự được gọi là anagram nếu ta có thể thay đổi thứ tự các kí tự trong chuỗi này để tạo thành chuỗi kia và ngược lại. Chẳng hạn các cặp chuỗi sau là anagram: ("taste", "state"), ("act", "cat"), ("listen", "slient"). Còn cặp chuỗi sau không phải là anagram: ("feel", "fell").

Còn cặp chuỗi sau không phải là anagram: ("computer", "poetry"). Tuy nhiên ta có thể xóa các kí tự 'c', 'm', 'u', ở chuỗi thứ nhất và kí tự 'y' ở chuỗi thứ hai để chúng trở thành cặp anagram ("opter", "poetr"). Như vậy ta cần 4 phép xóa.

**Yêu cầu:** Cho 2 chuỗi  $s_1$  và  $s_2$ . Tìm số kí tự ít nhất cần phải xóa để 2 chuỗi trở thành anagram.

**Dữ liệu:** Vào từ tập tin văn bản **ANADEL.INP**

- Dòng đầu tiên chứa chuỗi  $s_1$ .
- Dòng thứ hai chứa chuỗi  $s_2$ .
- Cả 2 chuỗi có độ dài không quá  $10^4$  các kí tự chữ cái latin in thường.

**Kết quả:** Ghi ra tập tin văn bản **ANADEL.OUT** một số nguyên là số kí tự ít nhất cần xóa.

**Ví dụ:**

ANADEL . INP	ANADEL . OUT
computer poetry	4

## CHUỖI PALINDROME

Cho chuỗi độ dài không quá  $10^6$  chỉ gồm các kí tự Latin in hoa. Hãy sắp xếp lại các kí tự của chuỗi sao cho chuỗi trở thành Palindrome.

**Dữ liệu:** Vào từ tập tin văn bản **PALSTR.INP** chứa chuỗi độ dài không quá  $10^6$  kí tự 'A', 'B', ... 'Z'.

**Kết quả:** Ghi ra tập tin văn bản **PALSTR.OUT** chuỗi Palindrome được hình thành từ các kí tự của chuỗi ban đầu. Nếu có nhiều kết quả thì ghi kết quả bất kỳ. Nếu không thể tạo thành chuỗi Palindrome thì ghi NO SOLUTION.

**Ví dụ:**

PALSTR . INP	PALSTR . OUT
AAAACACBA	AACABACAA

## KIỂM TRA PALINDROME

Chuỗi palindrome là chuỗi có giá trị giống nhau khi đọc từ trái sang hoặc từ phải sang, chẳng hạn chuỗi "abba". Một số chuỗi có thể biến thành palindrome nếu thay đổi thứ tự các kí tự trong chuỗi, một số chuỗi khác thì không. Chẳng hạn chuỗi "aabbcc" có thể tạo thành chuỗi palindrome "abccba", còn chuỗi "aaabcc" thì không thể.

**Yêu cầu:** Cho chuỗi  $s$  có độ dài không quá  $10^5$  kí tự chữ cái latin in thường. Hãy cho biết có thể thay đổi thứ tự các kí tự trong  $s$  để trở thành chuỗi palindrome.

**Dữ liệu:** Vào từ tập tin văn bản **CHECKPAL.INP** chứa chuỗi  $s$ .

**Kết quả:** Ghi ra tập tin văn bản **CHECKPAL.OUT** thông báo YES hoặc NO tương ứng với câu trả lời.

**Ví dụ:**

CHECKPAL . INP	CHECKPAL . OUT
aabbcc	YES
aaabcc	NO

**THUÊ PHÒNG**

Có  $n$  khách đăng ký thuê phòng khách sạn. Hiện khách sạn đang có  $m$  phòng trống. Mỗi khách đến thuê đều đặt ra yêu cầu về kích thước phòng mong muốn, nếu không đáp ứng được yêu cầu thì khách sẽ từ chối thuê. Yêu cầu của khách thứ  $i$  gồm có kích thước phòng  $a_i$  và biên độ  $k$ . Khi đó khách thứ  $i$  chỉ chấp nhận thuê các phòng có kích thước từ  $a_i - k$  đến  $a_i + k$ .

**Yêu cầu:** Cho danh sách các yêu cầu về kích thước của từng khách và kích thước của các phòng. Hãy tìm phương án phân phòng sao cho số khách thuê nhận được phòng theo đúng yêu cầu là nhiều nhất.

**Dữ liệu:** Vào từ tập tin văn bản **APPARTMENTS.INP**

- Dòng đầu tiên chứa 3 số nguyên  $n, m, k$  ( $1 \leq n, m \leq 2 \times 10^5$ ;  $0 \leq k \leq 10^9$ ) – số lượng khách thuê, số phòng trống và biên độ kích thước phòng mà khách thuê chấp nhận được.
- Dòng thứ hai chứa dãy gồm  $n$  số nguyên  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) – kích thước phòng thuê theo yêu cầu của từng khách.
- Dòng thứ ba chứa dãy gồm  $m$  số nguyên  $b_1, b_2, \dots, b_m$  ( $1 \leq b_i \leq 10^9$ ) – kích thước các phòng cho thuê.

**Kết quả:** Ghi ra tập tin văn bản **APPARTMENTS.OUT** một số nguyên là số lượng nhiều nhất các khách chấp nhận thuê phòng.

**Ví dụ:**

APPARTMENTS . INP	APPARTMENTS . OUT
4 3 5 60 45 80 60 30 60 75	2



## MÁY SẢN XUẤT

Một nhà máy có  $n$  máy sản xuất, các máy được đánh số từ 1 đến  $n$ . Máy thứ  $i$  cần  $t_i$  đơn vị thời gian để sản xuất ra một sản phẩm. Các máy sản xuất có thể hoạt động đồng thời và có thể được lên lịch để vận hành một cách tự do.

**Yêu cầu:** Tính thời gian tối thiểu để nhà máy sản xuất  $K$  sản phẩm.

**Dữ liệu:** Vào từ tập tin văn bản **MACHINES.INP**

- Dòng đầu tiên chứa 2 số nguyên  $n, K$  ( $1 \leq n \leq 2 \times 10^5$ ;  $1 \leq K \leq 10^9$ ) – số lượng máy sản xuất và số lượng sản phẩm cần sản xuất.
- Dòng tiếp theo chứa dãy gồm  $n$  số nguyên  $t_1, t_2, \dots, t_n$  ( $1 \leq t_i \leq 10^9$ ) – thời gian cần thiết để các máy tương ứng sản xuất ra một sản phẩm.

**Kết quả:** Ghi ra tập tin văn bản **MACHINES.OUT** một số nguyên là tổng thời gian tối thiểu để sản xuất ra  $K$  sản phẩm.

**Ví dụ:**

MACHINES . INP	MACHINES . OUT
3 7 3 2 5	8

**Giải thích:** máy 1 sản xuất 2 sản phẩm, máy 2 sản xuất 4 sản phẩm, máy 3 sản xuất 1 sản phẩm.

## MỨC THƯỞNG

Cho danh sách gồm  $n$  công việc. Công việc thứ  $i$  có thời gian xử lý  $t_i$  và thời hạn hoàn thành  $d_i$  (deadline). Nếu bạn thực hiện xong công việc thứ  $i$  tại thời điểm  $f_i$  thì bạn sẽ nhận được mức thưởng  $d_i - f_i$  (mức thưởng có thể bị âm, khi đó được xem là mức phạt). Thời điểm được tính bắt đầu từ 0 và bạn phải hoàn thành tất cả  $n$  công việc theo một thứ tự nào đó một cách tuần tự.

**Yêu cầu:** Hãy tính tổng mức thưởng tối đa có thể nhận được.

**Dữ liệu:** Vào từ tập tin văn bản **REWARD.INP**

- Dòng đầu tiên chứa số nguyên  $n$  ( $1 \leq n \leq 2 \times 10^5$ ) – số công việc được giao.
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo chứa 2 số nguyên  $t_i, d_i$  ( $1 \leq t_i, d_i \leq 10^6$ ) – mô tả một công việc.

**Kết quả:** Ghi ra tập tin văn bản **REWARD.OUT** tổng mức thưởng tối đa có thể nhận được.

**Ví dụ:**

REWARD . INP	REWARD . OUT
3 6 10 8 15 5 12	2

**PHÂN MÁY MÁY GIA CÔNG**

Một xưởng máy có 2 máy gia công sản phẩm, máy thứ nhất có hiệu suất làm việc nhanh gấp đôi máy thứ hai. Xưởng máy nhận đơn đặt hàng gồm  $n$  sản phẩm, sản phẩm thứ  $i$  có thời gian gia công  $t_i$ . Tại mỗi thời điểm, mỗi máy chỉ gia công 1 sản phẩm. Sau khi hoàn thiện một sản phẩm, máy sẽ được phân phối để gia công sản phẩm tiếp theo.

Thợ xưởng phân phối  $n$  sản phẩm cho 2 máy theo cách sau:

- Máy thứ nhất gia công bắt đầu từ sản phẩm thứ 1 trở đi (thứ tự thực hiện: 1, 2, ...).
- Máy thứ hai gia công bắt đầu từ sản phẩm thứ  $n$  trở lại (thứ tự thực hiện:  $n, n - 1, \dots$ ).
- Khi chỉ còn 1 sản phẩm chưa gia công mà cả 2 máy đều rồi thì phân cho máy thứ nhất thực hiện.

**Yêu cầu:** Cho biết số lượng sản phẩm mà mỗi máy gia công được theo cách phân phối trên.

**Dữ liệu:** Vào từ tập tin văn bản **DISTRIBUTION.INP**

- Dòng đầu tiên chứa số nguyên  $t (t \leq 100)$  – số bộ test.
- Dòng đầu tiên của mỗi bộ test chứa số nguyên  $n (1 \leq n \leq 10^5)$  – số sản phẩm.
- Dòng thứ hai của mỗi bộ test chứa dãy số nguyên  $t_1, t_2, \dots, t_n (1 \leq t_i \leq 10^9)$  – thời gian để gia công các sản phẩm tương ứng.

**Kết quả:** Ghi ra tập tin văn bản **DISTRIBUTION.OUT** gồm  $t$  dòng, mỗi dòng ghi 2 số nguyên  $x, y$  – tương ứng số lượng sản phẩm mà máy thứ nhất và máy thứ hai gia công được.

**Ví dụ:**

DISTRIBUTION . INP	DISTRIBUTION . OUT
3	4 1
5	1 3
2 6 2 1 7	3 2
4	
15 2 1 3	
5	
2 4 12 4 7	