**ĐỀ SỐ 27 - Ngày 13 - 10 - 2022**

**Bài 1. Xâu cân bằng**

Một xâu được gọi là **cân bằng** nếu nó thỏa mãn các điều kiện:

* Có độ dài n là một số nguyên dương chẵn.
* Mọi ký tự xuất hiện ở nửa bên trái (từ vị trí 1 đến vị trí n/2) cũng đều xuất hiện ở nửa bên phải (từ vị trí n/2 + 1 đến n).
* Mọi ký tự xuất hiện ở nửa bên phải cũng đều xuất hiện ở nửa bên trái.

(Lưu ý rằng ta đánh chỉ số cho các ký tự từ 1 đến n)

Cho xâu s dài không quá 105 ký tự, hãy lập trình kiểm tra xem nó có phải xâu cân bằng hay không.

**Input**

Dòng đầu chứa số nguyên dương T (T≤10) là số lượng câu hỏi.

Mỗi dòng trong T dòng tiếp theo chứa một xâu ss gồm không quá 105 chữ cái tiếng Anh in thường.

**Output**

In ra T dòng, mỗi dòng ghi YES nếu xâu tương ứng trong input là một xâu cân bằng, hoặc NO trong trường hợp ngược lại.

Ví dụ:

|  |  |
| --- | --- |
| **Balstr.inp** | **Balstr.out** |
| 4  baohieplpvip  abccba  anna  lqdoj | NO  YES  YES  NO |

**Bài 2.** **Tìm ước**

Một dãy B được gọi là *ước* của dãy A nếu như ghép liên tiếp một số nguyên lần dãy B ta thu được dãy A. Hãy tìm *ước* ít phần tử nhất của một dãy con đã cho

*Input:*

+Dòng đầu ghi *n* (*n*≤10000)

+Các dòng tiếp theo ghi *a1, a2, ..., an*

*Output:*

Một số nguyên duy nhất là số lượng phần tử của ước tìm được

Ví dụ:

|  |  |
| --- | --- |
| Timuoc.inp | Timuoc.out |
| 8  9 6 5 3 9 6 5 3 | 4 |

**Bài 3. Xóa k phần tử**

* Cho một mảng gồm N phần tử và số nguyên K(0≤K<N).
* Nhiệm vụ của chúng ta là xóa đi K phần tử từ mảng A sao cho số lượng phần tử còn lại khác nhau là nhiều nhất và in ra giá trị lớn nhất đó

**Input:**

* Dòng thứ nhất chứa số nguyên T - thể hiện số lượng testcase (1≤T≤100)
* T block tiếp theo ,mỗi block có dạng như sau:
  + Dòng thứ nhất chứa số nguyên N(0<N≤10000)
  + Dòng thứ hai chứa N số nguyên a1,a2,...,aN(1≤ai≤N)
  + Dòng thứ ba chứa số nguyên K(0≤K<N)

**Output:**

* Ứng với mỗi block, in ra đáp án cần tìm.

**Ví dụ:**

|  |  |
| --- | --- |
| **Xoakpt.inp** | **Xoakpt.out** |
| 1  3  1 1 2  1 | 2 |

**Giải thích:** Ta chỉ cần xóa đi 1 số 1 thì số phần tử khác nhau trong những phần tử còn lại lớn nhất là 2.

**Subtask:**

* 20% :0<N≤10
* 40% :11≤N≤100
* 40% :101≤N≤104

**Bài 4. Đổi số thành chữ**

Steve học ngoại ngữ. Bài tập hôm nay là học thuộc tên các chữ số (ở hệ cơ số 10). Để rèn luyện kỹ năng phản xạ nhanh, Steve viết một dãy ***n*** số nguyên dương ***a1***, ***a2***, . . ., ***an***, mỗi số không vượt quá 109 và không có các số 0 không có nghĩa ở đầu (1 ≤ ***n*** ≤ 1000). Với mỗi số Steve sẽ đọc chữ số có tên lớn nhất (theo thứ tự từ điển) trong số các tên xuất hiện trong số này.

Ví dụ, Steve học tiếng Pháp. Tên các chữ số trong tiếng Pháp là như sau:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|  | ***zero*** | ***un*** | ***deux*** | ***trois*** | ***quatre*** | ***Cinq*** | ***six*** | ***sept*** | ***huit*** | ***neuf*** |

Với số 908, chữ số mà Steve đọc là ***zero***.

***Yêu cầu***: Cho dãy 10 tên các chữ số từ 0 đến 9, số nguyên ***n*** và ***n*** số nguyên ***a1***, ***a2***, . . ., ***an***. Với mỗi số nguyên hãy nêu tên chữ số được đọc.

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

* Dòng đầu tiên chứa 10 xâu, xâu thứ ***i*** là tên chữ số ***i***, ***i*** = 0 ÷ 9, mỗi xâu không quá 50 ký tự, các xâu cách nhau một dấu cách,
* Dòng thứ 2 chứa số nguyên ***n***,
* Dòng thứ ***i*** trong ***n*** dòng tiếp theo chứa số nguyên ***ai***.

***Kết quả***: Đưa ra file văn bản NAMENUM.OUT ***n*** tên các chữ số được đọc, mỗi tên đưa ra trên một dòng, dòng thứ ***i*** xác định tên đọc trong số ***ai***.

***Ví dụ***:

|  |  |
| --- | --- |
| **NAMENUM.INP** | **NAMENUM.OUT** |
| zero un deux trois quatre cinq six sept huit neuf  3  123  456  908 | un  six  zero |

# Bài 5. Tính

Cho dãy số gồm số nguyên dương và một số nguyên . Người ta định nghĩa hai hàm số:

* số lượng cặp chỉ số sao cho
* .

**Yêu cầu:** Cho số Hãy tính .

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

* Dòng đầu gồm số
* Dòng tiếp theo gồm số cách nhau bởi ít nhất 1 dấu cách.

**Kết quả:** Ghi ra file văn bản FM.OUTgồm một số nguyên duy nhất là giá trị của .

**Ví dụ:**

|  |  |
| --- | --- |
| FM.INP | FM.OUT |
| 5 6  1 2 3 4 5 | 15 |

**Bài 6. Hàng cây**

Trong khu vườn, người ta trồng một hàng cây chạy dài gồm có N cây, mỗi cây có độ cao là ai, a2,.. .aN.

Người ta cần lấy M mét gỗ bằng cách đặt cưa máy sao cho lưỡi cưa ở độ cao H (mét) để cưa tất cả các cây có độ cao lớn hơn H (dĩ nhiên những cây có độ cao không lớn hơn H thì không bị cưa).

Ví dụ: Nếu hàng cây có các cây với độ cao tương ứng là 20; 15; 10 và 18 mét, cần lấy 7 mét gỗ. Lưỡi cưa đặt tại độ cao hợp lí là 15 mét thì độ cao của các cây còn lại sau khi bị cưa tương ứng là 15; 15; 10 và 15 mét. Tổng số mét gỗ lấy được là 8 mét (dư 1 mét).

Yêu cầu: Hãy tìm vị trí đặt lưỡi cưa hợp lí (số nguyên H lớn nhất) sao cho lấy được M mét gỗ và số mét gỗ dư ra là ít nhất.

Dữ liệu: Vào từ tệp văn bản WOOD.INP

Dòng thứ nhất chứa 2 số nguyên dương N và M (1<N<106; 1<M<2x109) cách nhau một dấu cách.

Dòng thứ hai chứa N số nguyên dương ai là độ cao của mỗi cây trong hàng (1<ai<109; i=1.. .N), mỗi số cách nhau ít nhất một dấu cách.

Kết quả. Đưa ra tệp WOOD.OUT là một số nguyên cho biết giá trị cần tìm.

|  |  |
| --- | --- |
| WOOD.INP | WOOD.OUT |
| 4 7  20 15 10 18 | 15 |

# Bài 7. Phủ

Cho n đoạn trên trục số, đoạn thứ i là [Li, Ri].

***Hãy chọn ra trong các đoạn kể trên một số ít nhất các đoạn để phủ hết đoạn [a, b]***

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

* Dòng 1: Chứa 3 số n, a, b
* n dòng tiếp theo, dòng thứ i chứa hai số Li và Ri

**Kết quả:** Ghi ra file văn bản COVER.OUT số k là số đoạn được chọn (Nếu không có cách chọn thì k = -1)

***Các số trên một dòng của Input/Output file cách nhau ít nhất một dấu cách***

**Ràng buộc: 1 ≤ n ≤ 100000; các số còn lại là số nguyên dương ≤ 30000; a ≤ b; ∀i: Li ≤ Ri**

**Ví dụ:**

|  |  |  |  |
| --- | --- | --- | --- |
| **COVER.INP** | **COVER.OUT** | **COVER.INP** | **COVER.OUT** |
| 8 2 10  4 8  1 3  2 3  1 4  3 4  7 10  9 11  8 11 | 3 | 8 1 200  1 4  2 5  4 5  6 45  6 7  5 7  100 200  50 99 | -1 |