

Buổi 12.

Bài 1. Cho một mảng gồm n ($3 \leq n \leq 100000$) số nguyên đôi một khác nhau, tìm và in ra 3 số lớn nhất trong mảng.

Ví dụ

Input

10

99 13 2 4 0 12 24 58 56 14

Output

99 58 56

Bài 2. Cho một mảng gồm n ($3 \leq n \leq 100000$) số nguyên đôi một khác nhau, liệt kê các phần tử trong mảng có ít nhất 2 phần tử khác lớn hơn nó.

Ví dụ

Input

6

1 5 4 3 2 7

Output

1 4 3 2

Bài 3. Cho một mảng gồm n ($3 \leq n \leq 100000$) số nguyên đôi một khác nhau, tìm và in ra 3 số nhỏ nhất trong mảng.

Ví dụ

Input

7

9 11 78 75 14 6 1

Output

9 6 1

Bài 4. Cho một mảng gồm n ($2 \leq n \leq 100000$) số nguyên dương tìm số lớn nhất và

lớn thứ hai trong mảng, trường hợp không có số lớn thứ 2 trong thì in -1.

Ví dụ

Input

7

1 2 3 4 5 6 6

Output

6 5

Input

7

99 99 99 99 99 99 99

Output

99 -1

Bài 5. Một số được định nghĩa là số đẹp nếu nó chứa cả chữ số 1 và chữ số 9. Cho một mảng số nguyên gồm n phần tử ($1 \leq n \leq 100$), đếm số lượng chữ số đẹp trong mảng.

Ví dụ

Input

5

79 19 91 99 129

Output

3

Bài 6. Tìm vị trí số chẵn cuối cùng trong mảng một chiều, nếu mảng không tồn tại số chẵn thì in ra -1.

Ví dụ

Input

6

7 8 1 2 2 3

Output

5

Bài 7. Tìm vị trí của số dương nhỏ nhất trong mảng 1 chiều, nếu mảng không tồn tại số dương thì in ra -1.

Ví dụ

Input

5-

1 -2 9 3 4

Output

4

Bài 8. Nhập vào mảng một chiều gồm n phần tử, và phần tử x, tìm phần tử trong mảng cách xa x nhất. Dữ liệu đầu vào dòng đầu tiên là số phần tử trong mảng, dòng thứ 2 là các phần tử trong mảng, dòng thứ 3 là giá trị của x.

Ví dụ

Input

5

1 2 3 -5 15

3

Output

15

Bài 9. Cho mảng một chiều gồm n số nguyên dương, tìm đoạn [a,b] chứa tất cả các phần tử xuất hiện trong mảng

Ví dụ

Input

5

1 9 10001 25 77

Output

1 10001

Bài 10. Cho mảng một chiều các số nguyên, liệt kê các phần tử có ít nhất một phần

tử liền kề trái dấu với nó

Ví dụ

Input

7-

1 2 3 -1 5 8 9

Output

-1 2 3 -1 5

Bài 11. Liệt kê các số trong mảng một chiều có chữ số đầu tiên của nó là chữ số lẻ.

Ví dụ

Input

5

17 88 23 14 90

Output

17 23 14 90

Bài 12. Liệt kê các số trong mảng một chiều các số nguyên chứa toàn chữ số lẻ.

Ví dụ

Input

10

1 2 3 4 55 666 7 898 1 4 23

Output

1 3 55 7

Bài 13. Tính tổng các số chính phương trong mảng một chiều các số nguyên không âm.

Ví dụ

Input

6

1 4 9 20 45 31

Output

14

Bài 14. Kiểm tra xem mảng cho trước có tồn tại 2 số nguyên liên tiếp hay không, nếu có in ra YES, ngược lại in ra NO.

Ví dụ

Input

5

8 14 99 15 20

Output

YES

Bài 15. Kiểm tra xem mảng cho trước có tăng dần hay không, mảng tăng dần được định nghĩa là mảng có phần tử đứng sau lớn hơn phần tử đứng trước nó. Nếu mảng tăng dần in ra YES, trường hợp ngược lại in ra NO.

Ví dụ

Input

9

1 2 3 4 5 6 6 7 9

Output

NO

Bài 16. Kiểm tra xem mảng cho trước có không tăng hay không, mảng không tăng được định nghĩa là mảng có phần tử đứng trước lớn hơn hoặc bằng phần tử đứng sau nó. Nếu mảng không tăng in ra YES, ngược lại in ra NO.

Ví dụ

Input

7

7 6 5 5 3 3 1

Output

YES

Bài 17. Đếm số lượng số nguyên tố trong mảng một chiều các số nguyên cho trước.

Ví dụ

Input

10

1 2 3 4 5 6 7 8 9 11

Output

5

Bài 18. Đếm số lượng số nguyên trong mảng có số lượng ước là số lẻ.

Ví dụ

Input

10

1 2 3 4 5 6 7 8 9 25

Output

4

Bài 19. Cho một dãy số nguyên dương có n phần tử ($2 \leq n \leq 50$). Hãy liệt kê số các phần tử trong dãy không nhỏ hơn các số đứng trước nó (tính cả phần tử đầu tiên).

Dữ liệu vào: Dòng 1 ghi số bộ test. Mỗi bộ test ghi ra 2 dòng: dòng 1 là số phần tử, dòng thứ 2 ghi dãy số. Kết quả ghi ra số phần tử thỏa mãn.

Ví dụ

Input

17

3 5 6 8 4 2 9

Output

5

Bài 20. Cho dãy số A có n phần tử chỉ bao gồm các số nguyên dương (không quá 105). Hãy đếm xem mỗi số xuất hiện bao nhiêu lần.

Dữ liệu vào: Dòng đầu tiên ghi số bộ test. Với mỗi bộ test: dòng đầu ghi số n (không quá 100); dòng tiếp theo

ghi n số của dãy.

Kết quả: Với mỗi bộ test ghi ra thứ tự bộ test, sau đó lần lượt là các số trong dãy theo thứ tự xuất hiện trong dãy và số lần xuất hiện của nó.

Ví dụ

Input

1

10

1 7 2 8 3 3 2 1 3 2

Output

Test 1:

1 xuất hiện 2 lần

7 xuất hiện 1 lần

2 xuất hiện 3 lần

8 xuất hiện 1 lần

3 xuất hiện 3 lần

Bài 21. Cho dãy số A có n phần tử chỉ bao gồm các số nguyên dương (không quá 105). Hãy xác định các số nguyên tố trong dãy và đếm xem mỗi số xuất hiện bao nhiêu lần.

Dữ liệu vào: Dòng đầu tiên ghi số bộ test. Với mỗi bộ test: dòng đầu ghi số n (không quá 100); dòng tiếp theo ghi n số của dãy.

Kết quả: Với mỗi bộ test ghi ra thứ tự bộ test, sau đó lần lượt là các số nguyên tố trong dãy theo thứ tự từ nhỏ đến lớn và số lần xuất hiện của nó.

Ví dụ

Input

2

10

1 7 2 8 3 3 2 1 3 2

5

1 3 5 5 13

Output

Test 1:

2 xuất hiện 3 lần

3 xuất hiện 3 lần

7 xuất hiện 1 lần

Test 2:

3 xuất hiện 1 lần

5 xuất hiện 2 lần

13 xuất hiện 1 lần

Bài 22. Khi chuẩn bị một giải đấu, các điều phối viên của Codeforces cố gắng tốt nhất để

làm cho vấn đề đầu tiên dễ dàng nhất có thể. Lần này, điều phối viên đã chọn một số vấn đề và hỏi mọi người về ý kiến của họ. Mỗi người trả lời vấn đề này là dễ hay khó.

Nếu ít nhất một trong số n người này trả lời rằng vấn đề là khó khăn, điều phối viên quyết định thay đổi vấn đề. Đối với các phản hồi nhất định, kiểm tra xem vấn đề có đủ dễ dàng không.

Input

Dòng đầu tiên chứa một số nguyên n ($1 \leq n \leq 100$) - số người được yêu cầu đưa ra ý kiến của họ.

Dòng thứ hai chứa n số nguyên, mỗi số nguyên là 0 hoặc 1. Nếu số nguyên i là 0, thì người thứ i nghĩ rằng vấn đề là dễ dàng; nếu là 1, thì người thứ i nghĩ rằng vấn đề thật khó khăn.

Output

In một từ: "EASY" nếu vấn đề dễ dàng theo tất cả các phản hồi hoặc "HARD" nếu

có ít nhất một người cho rằng vấn đề khó khăn.

Ví dụ

Input

3

0 0 1

Output

HARD

Submit tại: <https://codeforces.com/problemset/problem/1030/A>

Bài 23. Bạn được cung cấp n chip trên một dòng số. Chip thứ i được đặt tại tọa độ nguyên

x_i . Một số chip có thể có tọa độ bằng nhau.

Bạn có thể thực hiện một trong hai loại di chuyển sau bất kỳ số lần (có thể là 0) trên bất kỳ chip nào:

Di chuyển chip i sang trái 2 đơn vị hoặc 2 đơn vị sang phải miễn phí (nghĩa là thay thế tọa độ x_i hiện tại bằng $x_i - 2$ hoặc bằng $x_i + 2$);

Di chuyển chip i sang trái hoặc sang phải 1 đơn vị và trả một xu cho lần di chuyển này (tức là thay thế tọa độ hiện tại x_i bằng $x_i - 1$ hoặc bằng $x_i + 1$).

Lưu ý rằng được phép di chuyển chip sang bất kỳ tọa độ nguyên nào, bao gồm âm và 0.

Nhiệm vụ của bạn là tìm ra tổng số xu tối thiểu cần thiết để di chuyển tất cả n chip vào cùng một tọa độ (nghĩa là tất cả x_i phải bằng nhau sau một số lần di chuyển).

Input

Dòng đầu tiên của đầu vào chứa một số nguyên n ($1 \leq n \leq 100$) - số lượng chip.

Dòng thứ hai của đầu vào chứa n số nguyên x_1, x_2, \dots, x_n ($1 \leq x_i \leq 10^9$), trong đó x_i là tọa độ của chip thứ i .

Output

In một số nguyên - tổng số xu tối thiểu cần thiết để di chuyển tất cả n chip đến cùng tọa độ.

Ví dụ

Input

3

1 2 3

Output

1

Submit tại: <https://codeforces.com/problemset/problem/1213/A>

Bài 24. ZS the Coder đang mã hóa trên một máy tính điện. Nếu bạn không gõ một từ trong

một giây liên tiếp, mọi thứ bạn gõ sẽ biến mất!

Nếu bạn gõ một từ ở giây thứ a và sau đó là từ tiếp theo ở giây b , thì nếu $b - a \leq c$, chỉ từ mới được thêm vào các từ khác trên màn hình. Nếu $b - a > c$, thì mọi thứ trên màn hình sẽ biến mất và sau đó từ bạn đã gõ sẽ xuất hiện trên màn hình.

Ví dụ: nếu $c = 5$ và bạn đã gõ các từ ở giây 1, 3, 8, 14, 19, 20 thì ở giây thứ 8 sẽ có 3 từ trên màn hình. Sau đó, mọi thứ biến mất vào giây thứ 13 vì không có gì được gõ. Ở giây 14 và 19, hai từ khác được gõ và cuối cùng, ở 20 giây thứ hai, một từ nữa được gõ và tổng cộng 3 từ vẫn còn trên màn hình

Bạn được cung cấp thời gian khi ZS gõ các từ. Xác định có bao nhiêu từ vẫn còn trên màn hình sau khi anh ta gõ xong mọi thứ.

Input

Dòng đầu tiên chứa hai số nguyên n và c ($1 \leq n \leq 100\,000$, $1 \leq c \leq 10^9$) - số lượng từ ZS mà Coder đã nhập và độ trễ của máy tính điện tương ứng.

Dòng tiếp theo chứa n số nguyên t_1, t_2, \dots, t_n ($1 \leq t_1 < t_2 < \dots < t_n \leq 10^9$), trong đó t_i biểu thị số thứ hai khi ZS gõ từ thứ i .

Output

In một số nguyên dương duy nhất, số lượng từ còn lại trên màn hình sau khi tất cả n từ được gõ.

Ví dụ

Input

6 5

1 3 8 14 19 20

Output

3

Bài 25.

Nhập vào mảng số nguyên gồm n ($1 \leq n \leq 1000$) phần tử, sắp xếp và in ra các phần tử trong mảng theo thứ tự tăng dần. Chú ý cài đặt bằng cả 2 thuật toán (selection và interchange sort).

Input

Dòng đầu tiên là số phần tử của mảng.

Dòng thứ 2 là các phần tử trong mảng.

Output

Các phần tử trong mảng theo thứ tự tăng dần.

Ví dụ

Input

5

8 2 4 5 1

Output

1 2 4 5 8

Bài 26.

Nhập vào mảng số nguyên gồm n ($1 \leq n \leq 1000$) phần tử, sắp xếp và in ra các phần tử trong mảng theo thứ tự giảm dần. Chú ý cài đặt bằng cả 2 thuật toán (selection và interchanger sort).

Input

Dòng đầu tiên là số phần tử của mảng.

Dòng thứ 2 là các phần tử trong mảng.

Output

Các phần tử trong mảng theo thứ tự tăng dần.

Ví dụ

Input

5

8 2 4 5 1

Output

8 5 4 2 1

Bài 27. Interchange sort.

Hãy thực hiện thuật toán sắp xếp đổi chỗ trực tiếp trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

Input

Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

Output

Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

Ví dụ

Input

4

5 7 3 2

Output

Buoc 1: 2 7 5 3

Buoc 2: 2 3 7 5

Buoc 3: 2 3 5 7

Bài 28. Selection sort.

Hãy thực hiện thuật toán sắp xếp chọn trên dãy N số nguyên. Ghi ra các bước thực hiện thuật toán.

Input

Dòng 1 ghi số N (không quá 100). Dòng 2 ghi N số nguyên dương (không quá 100).

Output

Ghi ra màn hình từng bước thực hiện thuật toán. Mỗi bước trên một dòng, các số trong dãy cách nhau đúng một khoảng trống.

Ví dụ

Input

4

5 7 3 2

Output

Buoc 1: 2 7 3 5

Buoc 2: 2 3 7 5

Buoc 3: 2 3 5 7