

[Comparator_Binary Search]. Bài 1. Comparator

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Bài 1. Cho mảng $A[]$ gồm N phần tử, bạn hãy sắp xếp các phần tử trong mảng theo yêu cầu như sau :

- Sắp xếp số lượng chữ số chẵn xuất hiện trong số tăng dần, nếu 2 số còn cùng số lượng chữ số chẵn thì số nhỏ hơn in trước.
- Sắp xếp theo số lượng chữ số lẻ xuất hiện trong số ban đầu tăng dần, nếu 2 số có cùng số lượng chữ số lẻ thì số nào xuất hiện trước sẽ in ra trước. (Bài này các bạn làm theo mảng và vector để luyện tập thêm)

Input Format

- Dòng 1 là N : số lượng phần tử trong mảng
- Dòng 2 là N số trong mảng

Constraints

- $1 \leq N \leq 10^5$
- $0 \leq A[i] \leq 10^9$

Output Format

- Dòng 1 in ra các phần tử sau khi sắp xếp theo yêu cầu 1
- Dòng 2 in ra các phần tử sau khi sắp xếp theo yêu cầu 2

Sample Input 0

```
12
4 9 5 7 15 4 13 15 1 8 4 15
```

Sample Output 0

```
1 5 7 9 13 15 15 15 4 4 4 8
4 4 8 4 9 5 7 1 15 13 15 15
```

[Comparator_Binary Search]. Bài 2.

Sắp xếp theo trị tuyệt đối

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng $A[]$ gồm N phần tử và số nguyên dương X , bạn hãy sắp xếp các phần tử trong mảng theo yêu cầu như sau :

- Sắp xếp các phần tử trong mảng theo abs của hiệu của các phần tử này với X tăng dần (tính trị tuyệt đối của $A[i]$ với X rồi sắp tăng dần). Nếu có 2 phần tử có cùng giá trị sắp xếp này thì số nào nhỏ hơn sẽ in trước.
- Sắp xếp so cho các phần tử là số chẵn đứng trước, các phần tử là số lẻ đứng sau, trong đó chẵn tăng dần, lẻ giảm dần với 1 comparator duy nhất, không tách mảng.

```
#include <bits/stdc++.h>
#include <algorithm>
using namespace std;

using ll = long long;

int X;

bool cmp1(int x, int y){
    if(abs(x - X) != abs(y - X)){
        return abs(x - X) < abs(y - X);
    }
    else{
        return x < y;
    }
}

bool cmp2(int x, int y){
    if(x % 2 == 0 && y % 2 == 0){
        return x < y;
    }
    if(x % 2 == 1 && y % 2 == 1){
        return x > y;
    }
    if(x % 2 == 0 && y % 2 == 1){
        return true;
    }
    return false;
}
```

```

}

int main(){
    int n; cin >> n >> X;
    int a[n];
    for(int i = 0; i < n; i++){
        cin >> a[i];
    }
    sort(a, a + n, cmp1);
    for(int x : a) cout << x << ' ';
    cout << endl;
    sort(a, a + n, cmp2);
    for(int x : a) cout << x << ' ';
}

```

Input Format

- Dòng 1 là N và X : số lượng phần tử trong mảng và số X
- Dòng 2 là N số trong mảng

Constraints

- $1 \leq N \leq 10^5$
- $0 \leq A[i], X \leq 10^9$

Output Format

- Dòng 1 in ra các phần tử sau khi sắp xếp theo yêu cầu 1
- Dòng 2 in ra các phần tử sau khi sắp xếp theo yêu cầu 2

Sample Input 0

```

14 26
83 86 49 15 31 81 33 10 77 38 49 86 71 28

```

Sample Output 0

```

28 31 33 15 38 10 49 49 71 77 81 83 86 86
10 28 38 86 86 83 81 77 71 49 49 33 31 15

```

[Sắp xếp - Tìm Kiếm]. Bài 1. Các số khác nhau trong mảng

- [Problem](#)
- [Submissions](#)

- [Discussions](#)

Cho mảng số nguyên gồm N phần tử. Hãy đếm xem mảng có bao nhiêu phần tử riêng biệt

Input Format

Dòng đầu tiên là N Dòng thứ 2 là các phần tử trong mảng a_1, a_2, \dots, a_N

Constraints

$$1 \leq n \leq 2 \cdot 10^5; 1 \leq a_i \leq 10^9$$

Output Format

In ra số lượng phần tử riêng biệt trong mảng.

Sample Input 0

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

Sample Output 0

```
5
```

Explanation 0

5 phần tử khác nhau trong mảng là 1, 2, 3, 4, 5

[Sắp xếp - Tìm Kiếm]. Bài 1.2. Tăng dần, giảm dần

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng số nguyên $A[]$ có N phần tử, hãy sắp xếp các phần tử trong mảng theo thứ tự tăng dần, sau đó sắp xếp các phần tử theo thứ tự giảm dần.

Input Format

Dòng đầu tiên là số nguyên dương N. Dòng thứ 2 là N phần tử trong mảng, các phần tử viết cách nhau một dấu cách.

Constraints

$1 \leq N \leq 10^5$; $-10^9 \leq A[i] \leq 10^9$

Output Format

Dòng đầu tiên in ra các phần tử trong mảng theo thứ tự tăng dần. Dòng thứ 2 in ra các phần tử trong mảng theo thứ tự giảm dần.

Sample Input 0

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

Sample Output 0

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

[Sắp xếp - Tìm Kiếm]. Bài 2. Sắp xếp theo trị tuyệt đối

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng số nguyên $A[]$ có N phần tử, hãy sắp xếp các phần tử trong mảng theo thứ tự giá trị tuyệt đối tăng dần. Nếu có 2 số có cùng giá trị tuyệt đối thì số nào xuất hiện trước sẽ được in ra trước

Input Format

Dòng đầu tiên là số nguyên dương N . Dòng thứ 2 là N phần tử trong mảng, các phần tử viết cách nhau một dấu cách.

Constraints

$1 \leq N \leq 10^5$; $-10^9 \leq A[i] \leq 10^9$

Output Format

In ra các phần tử trong mảng sau khi sắp xếp theo thứ tự tăng dần

Sample Input 0

```
5
1 -3 2 -5 4
```

Sample Output 0

```
1 2 -3 4 -5
```

[Sắp xếp - Tìm Kiếm]. Bài 3. Sắp xếp theo tổng chữ số

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng số nguyên $A[]$ có N phần tử, hãy sắp xếp các phần tử trong mảng theo thứ tự tổng chữ số tăng dần, nếu 2 số có cùng tổng chữ số, thì số nào nhỏ hơn sẽ được in trước.

Input Format

Dòng đầu tiên là số nguyên dương N . Dòng thứ 2 là N phần tử trong mảng, các phần tử viết cách nhau một dấu cách.

Constraints

$1 \leq N \leq 10^5$; $0 \leq A[i] \leq 10^9$

Output Format

In ra các phần tử trong mảng sau khi sắp xếp

Sample Input 0

```
5
999976710 999982875 999974431 999984407 999972533
```

Sample Output 0

```
999974431 999972533 999976710 999984407 999982875
```

[Sắp xếp - Tìm Kiếm]. Bài 4. Khoảng cách nhỏ nhất

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng số nguyên $A[]$ có N phần tử, tìm độ chênh lệch nhỏ nhất giữa 2 phần tử $A[i]$ và $A[j]$ trong mảng (i và j khác nhau).

Input Format

Dòng đầu tiên là số nguyên dương N . Dòng thứ 2 là N phần tử trong mảng, các phần tử viết cách nhau một dấu cách.

Constraints

$1 \leq N \leq 10^6$; $0 \leq A[i] \leq 10^9$

Output Format

In ra độ lệch nhỏ nhất của 2 phần tử trong mảng

Sample Input 0

```
5
1 2 7 9 0
```

Sample Output 0

```
1
```

[Sắp xếp - Tìm Kiếm]. Bài 5. Số xuất hiện nhiều nhất trong mảng

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng số nguyên $A[]$ có N phần tử, hãy tìm số xuất hiện nhiều nhất trong mảng.

Input Format

Dòng đầu tiên là số nguyên dương N . Dòng thứ 2 là N phần tử trong mảng, các phần tử viết cách nhau một dấu cách.

Constraints

$1 \leq N \leq 10^5$; $-10^9 \leq A[i] \leq 10^9$

Output Format

In ra số có số lần xuất hiện nhiều nhất và tần suất tương ứng, nếu có nhiều số có cùng số lần xuất hiện thì in ra số nhỏ nhất.

Sample Input 0

```
5
1 2 2 1 3
```

Sample Output 0

```
1 2
```

[Sắp xếp - Tìm Kiếm]. Bài 11. Trộn 2 dãy

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

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

Constraints

$1 \leq n, m \leq 10^6$; $0 \leq b[i], c[i] \leq 10^6$

Output Format

In ra dãy là thứ tự trộn, trong trường hợp trong 2 mảng b và c có phần tử có cùng giá trị t in ra phần tử ở mảng b trước.

Sample Input 0

```
5 5
4760 9724 9798 20124 25974
3397 9166 13054 18273 30455
```

Sample Output 0

```
c1 b1 c2 b2 b3 c3 c4 b4 b5 c5
```

[Sắp xếp - Tìm Kiếm]. Bài 9. Tìm kiếm nhị phân

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng số nguyên $A[]$ có N phần tử đã được sắp xếp theo thứ tự giảm dần. Có Q truy vấn, mỗi truy vấn yêu cầu bạn kiểm tra xem phần tử X có xuất hiện trong mảng hay không?

Input Format

Dòng đầu tiên là số nguyên dương N . Dòng thứ 2 là N phần tử trong mảng, các phần tử viết cách nhau một dấu cách. Dòng thứ 3 là số lượng truy vấn Q . Q dòng tiếp theo mỗi dòng là một số nguyên dương X .

Constraints

$1 \leq N \leq 10^6$; $1 \leq Q \leq 10^3$; $0 \leq A[i], X \leq 10^9$

Output Format

Đối với truy vấn in ra YES trên 1 dòng nếu X xuất hiện trong mảng, ngược lại in ra NO.

Sample Input 0

```
5
5 4 3 2 1
2
2
8
```

Sample Output 0

```
YES
NO
```

[Sắp xếp - Tìm Kiếm]. Bài 12. Khiêu vũ

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

Input: gồm 3 dòng - Dòng thứ nhất là hai số n, m mỗi số cách nhau một khoảng trắng. - Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n là chiều cao các bạn nam. - Dòng thứ ba gồm m số nguyên b_1, b_2, \dots, b_m là chiều cao các bạn nữ.

Constraints

$1 \leq n, m \leq 10^5$; $1 \leq a[i], b[i] \leq 10^6$;

Output Format

In ra số lượng cặp khiêu vũ nhiều nhất ghép được.

Sample Input 0

```
5 5
2668 2956 20933 21199 24224
11521 13084 19573 25628 28958
```

Sample Output 0

```
3
```

[Sắp xếp - Tìm Kiếm]. Bài 13. Xếp gạch

- [Problem](#)

- [Submissions](#)
- [Discussions](#)

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 Format

- Dòng đầu tiên là số nguyên n - là số viên gạch.
- Dòng tiếp theo gồm n số nguyên a_1, a_2, \dots, a_n mỗi số cách nhau một khoảng trắng.

Constraints

$$1 \leq n \leq 10^5; 0 \leq a_i \leq 10^6$$

Output Format

Số nguyên xác định chiều cao cao nhất của chồng gạch mà Nam sắp được.

Sample Input 0

```
4
1 2 3 4
```

Sample Output 0

```
4
```

[Sắp xếp - Tìm Kiếm]. Bài 14. Vắt sữa bò

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

- Dòng thứ nhất là số nguyên là số lượng con bò.
- Dòng thứ hai gồm n số nguyên a_1, a_2, \dots, a_n là sản lượng sữa của các con bò.

Constraints

$1 \leq n \leq 10^5; 1 \leq a[i] \leq 10^6$

Output Format

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

Sample Input 0

```
4
4 4 4 4
```

Sample Output 0

```
10
```

[Sắp xếp - Tìm Kiếm]. Bài 15. The 2014 ACM-ICPC Asia Jakarta Regional Contest

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

- Dòng đầu chứa 2 số nguyên dương N, K ;
- Dòng thứ hai chứa N số nguyên dương - các phần tử của tập hợp

Constraints

$1 \leq N \leq 10^5$; $1 \leq K \leq 10^6$; Các phần tử trong tập hợp là số nguyên có trị tuyệt đối không vượt quá 10^6

Output Format

Kết quả của bài toán

Sample Input 0

```
7 1
2 6 1 7 3 4 9
```

Sample Output 0

```
3
```

Sample Input 1

```
8 10
100 200 300 400 500 600 700 800
```

Sample Output 1

```
8
```

[Sắp xếp - Tìm Kiếm]. Bài 17. Cặp số có tổng bằng K

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng a gồm n phần tử và số nguyên dương k . Đếm số lượng cặp số a_i, a_j ($i \neq j$) có tổng bằng k .

Gợi ý : Sắp xếp mảng tăng dần sau đó với mỗi phần tử $a[i]$ trong mảng tìm xem trong đoạn $[i + 1, n - 1]$ có bao nhiêu phần tử có giá trị là $k - a[i]$, bằng cách tìm vị trí đầu tiên và vị trí cuối cùng của phần tử có giá trị là $k - a[i] \Rightarrow$ Số lượng

Input Format

Dòng thứ 1 là số lượng phần tử trong mảng và số nguyên dương k ; Dòng thứ 2 là n phần tử trong mảng

Constraints

$2 \leq n \leq 10^6$; $1 \leq k \leq 10^6$; $0 \leq a[i] \leq 10^6$;

Output Format

In ra số lượng cặp số có tổng bằng k

Sample Input 0

```
4 4
2 2 2 2
```

Sample Output 0

```
6
```

[Sắp xếp - Tìm Kiếm]. Bài 18. Cặp số có tổng nhỏ hơn K

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng a gồm n phần tử và số nguyên dương k . Đếm số lượng cặp số a_i, a_j ($i \neq j$) có tổng nhỏ hơn k .

Input Format

Dòng thứ 1 là số lượng phần tử trong mảng và số nguyên dương k ; Dòng thứ 2 là n phần tử trong mảng

Constraints

$2 \leq n \leq 10^6$; $1 \leq k \leq 10^6$; $0 \leq a[i] \leq 10^6$;

Output Format

In ra số lượng cặp số có tổng nhỏ hơn k

Sample Input 0

```
4 5
```

2 2 2 2

Sample Output 0

6

[Sắp xếp - Tìm Kiếm]. Bài 19. Cặp số có tổng lớn hơn k

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho mảng a gồm n phần tử và số nguyên dương k . Đếm số lượng cặp số a_i, a_j ($i \neq j$) có tổng lớn hơn k .

Gợi ý : Sử dụng `binary_search`, đối với mỗi phần tử $a[i]$ đếm số lượng phần tử trong mảng (đã sắp xếp) $> k - a[i]$ bằng cách tìm vị trí đầu tiên của phần tử $> k - a[i]$, từ vị trí này tới cuối dãy sẽ đều là các phần tử $> k - a[i]$

Input Format

Dòng thứ 1 là số lượng phần tử trong mảng và số nguyên dương k ; Dòng thứ 2 là n phần tử trong mảng

Constraints

$2 \leq n \leq 10^6$; $1 \leq k \leq 10^6$; $0 \leq a[i] \leq 10^6$;

Output Format

In ra số lượng cặp số có tổng lớn hơn k

Sample Input 0

4 5
2 3 4 5

Sample Output 0

5

[Sắp xếp - Tìm Kiếm]. Bài 20. Điền số còn thiếu

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

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.

Constraints

$1 \leq n \leq 10^6$; $0 \leq a[i] \leq 10^6$;

Output Format

In ra số lượng số còn thiếu

Sample Input 0

```
5
4 5 3 8 6
```

Sample Output 0

```
1
```

[Sắp xếp - Tìm Kiếm]. Bài 21. Hợp và giao của 2 mảng

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

Cho 2 mảng số nguyên, thực hiện tìm hợp và giao của 2 mảng. Các phần tử trong mỗi mảng khác nhau đôi một.

Input Format

Dòng đầu tiên là số lượng phần tử của 2 dãy n và m . Dòng thứ 2 là n phần tử trong dãy số 1. Dòng thứ 3 là m phần tử trong dãy thứ 2.

Constraints

$1 \leq n, m \leq 10^5$; $-10^6 \leq a[i] \leq 10^6$;

Output Format

Dòng đầu tiên in ra giao của 2 mảng. Dòng thứ 2 in ra hợp của 2 mảng

Sample Input 0

```
3 5
1 2 3
1 2 9 5 3
```

Sample Output 0

```
1 2 3
1 2 3 5 9
```

[Sắp xếp - Tìm Kiếm]. Bài 22. Sắp xếp chữ số

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

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.

Constraints

$1 \leq n \leq 10^5$; $0 \leq A[i] \leq 10^6$;

Output Format

In ra các chữ số xuất hiện trong các số của mảng A theo thứ tự tăng dần

Sample Input 0

```
3
976117 988196 993766
```

Sample Output 0

```
1 3 6 7 8 9
```

[Sắp xếp - Tìm Kiếm]. Bài 24. Biểu thức nhỏ nhất

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

Dòng đầu chứa hai số nguyên dương n, k ; Dòng thứ hai chứa n số nguyên không âm a_1, a_2, \dots, a_n ;

Constraints

$1 \leq k < n \leq 10^5$; $0 \leq a[i] \leq 10^6$

Output Format

In ra giá trị lớn nhất của biểu thức

Sample Input 0

```
6 3
9560 5571 9008 3649 1473 3782
```

Sample Output 0

```
22799
```

[Sắp xếp - Tìm Kiếm]. Bài 25. Check in sân bay

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

Dòng đầu tiên là số nguyên dương N ; 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 .

Constraints

$1 \leq N \leq 10^5$; $1 \leq T[i], D[i] \leq 10^6$

Output Format

In ra đáp án tìm được.

Sample Input 0

```
3
2 1
8 3
5 7
```

Sample Output 0

```
15
```

[Sắp xếp - Tìm Kiếm]. Bài 30. Đèn lồng

- [Problem](#)
- [Submissions](#)

- [Discussions](#)

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 Format

Dòng đầu tiên chứa hai số nguyên n, l ($1 \leq n \leq 10^5, 1 \leq l \leq 10^9$) - số lượng đèn lồng và chiều dài đường phố tương ứng. Dòng tiếp theo chứa n số nguyên a_i ($0 \leq a_i \leq l$). Nhiều đèn lồng có thể được đặt tại cùng một điểm. Đèn lồng có thể nằm ở cuối phố.

Constraints

$1 \leq n \leq 10^5, 1 \leq l \leq 10^9; 0 \leq a_i \leq l;$

Output Format

In bán kính ánh sáng tối thiểu d , cần thiết để chiếu sáng cả đường phố. In ra kết quả với độ chính xác là 10 số sau dấu phẩy

Sample Input 0

```
7 15
15 5 3 7 9 14 0
```

Sample Output 0

```
2.5000000000
```

[Sắp xếp - Tìm Kiếm]. Bài 26. Tìm kiếm cặp số có hiệu bằng X

- [Problem](#)
- [Submissions](#)
- [Discussions](#)

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 Format

Dòng thứ nhất là cặp số N, X ; Dòng tiếp theo là N số $A[i]$ là các phần tử của mảng $A[]$.

Constraints

$1 \leq N \leq 10^5$; $1 \leq X, A[i] \leq 10^5$.

Output Format

In ra 1 nếu tìm thấy một cặp số có hiệu bằng X , ngược lại in ra -1.

Sample Input 0

```
5 3
1 1 2 3 5
```

Sample Output 0

```
1
```