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

* [**Problem**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/sap-xep-tim-kiem-bai-2-sap-xep-theo-tr-tuyet-doi)
* [**Submissions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/sap-xep-tim-kiem-bai-2-sap-xep-theo-tr-tuyet-doi/submissions)
* [**Discussions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/sap-xep-tim-kiem-bai-2-sap-xep-theo-tr-tuyet-doi/forum)

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<=N<=10^5; -10^9<=A[i]<=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

**[Comparator\_Binary Search]. Bài 2. Sắp xếp theo trị tuyệt đối**

* [**Problem**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-2-sap-xep-theo-tr-tuyet-doi)
* [**Submissions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-2-sap-xep-theo-tr-tuyet-doi/submissions)
* [**Discussions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-2-sap-xep-theo-tr-tuyet-doi/forum)

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<=N<=10^5
* 0<=A[i], X<=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

**[Comparator\_Binary Search]. Bài 1. Comparator**

* [**Problem**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-1-comparator-1)
* [**Submissions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-1-comparator-1/submissions)
* [**Discussions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-1-comparator-1/forum)

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<=N<=10^5
* 0<=A[i]<=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 4. Pair sort**

* [**Problem**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-4-pair-sort)
* [**Submissions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-4-pair-sort/submissions)
* [**Discussions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-4-pair-sort/forum)

Cho mảng A[] gồm N điểm trong hệ tọa độ Oxy, hãy sắp xếp các điểm này theo khoảng cách về gốc tọa độ tăng dần, nếu 2 điểm có cùng khoảng cách tới gốc tọa độ thì in ra theo hoành độ tăng dần, nếu tiếp tục 2 điểm này có cùng hoành độ thì in ra theo thứ tự tung độ tăng dần.

**Input Format**

* Dòng đầu tiên là N : số lượng điểm
* N dòng tiếp theo mỗi dòng là 2 số nguyên x, y tương ứng với hoành độ và tung độ

**Constraints**

* 1<=N<=10^5;
* -1000<=x,y<=1000;

**Output Format**

In ra các điểm sau khi sắp xếp

**Sample Input 0**

13

-42 -76

47 43

12 7

-62 31

7 64

42 -92

-89 60

45 41

3 54

-41 40

20 -24

88 42

0 12

**Sample Output 0**

0 12

12 7

20 -24

3 54

-41 40

45 41

47 43

7 64

-62 31

-42 -76

88 42

42 -92

-89 60

**[Comparator\_Binary Search]. Bài 5. Sort pair 2**

* [**Problem**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-5-sort-pair-2)
* [**Submissions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-5-sort-pair-2/submissions)
* [**Discussions**](https://www.hackerrank.com/contests/c-t2-2023-sort-va-comparison-function/challenges/comparator-binary-search-bai-5-sort-pair-2/forum)

Cho các điểm trong hệ tọa độ OXYZ, nhiệm vụ của bạn là sắp xếp các điểm này theo hoành độ tăng dần, nếu có cùng hoành độ thì sắp theo tung độ tăng dần, nếu có cùng tùng độ tiếp tục sắp xếp theo cao độ tăng dần. Sử dụng pair như sau để lưu các tọa độ này :

pair<int, pair<int, int>>

**Input Format**

* Dòng đầu tiên là N : số lượng điểm.
* N dòng tiếp theo mỗi dòng là 3 số nguyên x, y, z tương ứng với 1 điểm

**Constraints**

* 1<=N<=10^5
* x, y, z là số nguyên dương không quá 1000

**Output Format**

In ra các điểm sau khi sắp xếp

**Sample Input 0**

12

37 4 54

37 25 75

88 46 30

40 40 90

87 99 80

73 76 49

20 95 25

52 3 8

65 79 56

88 65 87

51 50 12

87 29 44

**Sample Output 0**

20 95 25

37 4 54

37 25 75

40 40 90

51 50 12

52 3 8

65 79 56

73 76 49

87 29 44

87 99 80

88 46 30

88 65 87