CS F211
Data Structures and Algorithms
Assignment - 7

Non-comparative sorting and Heaps

Allowed Language: **C**

February 21, 2024

General Tips

- **Note: You have to use Heaps to solve problems G-J, explicit sorting is not allowed.**

- Try to use functions as much as possible in your code. Functions increase reusability and the pass-by-value feature provides a significant help sometimes. Modularizing your code also helps you to debug efficiently.

- Use `scanf` to read characters/strings from STDIN. Avoid using `getchar`, `getc` or `gets`. Try to read up about character suppression in `scanf` as it will be very helpful in some of the problems.

- Use `printf` instead of `putc`, `putchar` or `puts` to print character/string output on STDOUT.

- Indent your code appropriately and use proper variable names. These increase readability and writability of the code. Also, Use comments wherever necessary.

- Use a proper IDEs like Sublime Text or VSCode as they help to run and test your code on multiple test-cases easily.

- **Note:** Kindly try to do all of these questions by yourself at least once. Spend some time thinking about it, or trying to code it instead of directly asking help of your friends or searching it up online. This helps you understand the question, allowing you to solve further questions which are not in the scope of this Assignment yourself.

A: Algorithme de Radix Sort

Radix sort is a non-comparative sorting algorithm. It uses a stable sorting algorithm to sort the elements starting from least significant digits first. The array would be sorted after all the digits have been processed. Radix sort can be used to sort numbers of any radix system, and we want you to use it for Hexadecimal numbers. In Hexadecimal system, each digit can take a value from 0-15, where 10-15 are represented with A-F. (0123456789ABCDEF). The decimal equivalent of the Hexadecimal number would be $\Sigma_{i=0}^{len-1} 16^i * d_i$ ($i = 0$ is rightmost, also called LSB).
(Refer to Wiki if any doubts)

Given an array of size n, find the state of the array after the k-th digit is processed (Instead of processing every digit, stop at k-th digit, assuming 1-based indexing).

Input

The first line contains two positive integers n k ($1 \le n \le 10^5, 1 \le k \le 10$), the size of array a and number of digits to process.
The second line contains n Hexadecimal numbers $a_1, a_2, \ldots, a_n$ ($1 \le len(a_i) \le 100$), a_i contains only 1-9 and A-F

Output

The only line of output should contain n Hexadecimal numbers - a after processing k-th digit.

```
input
4 1
3 A 5 2

output
2 3 5 A
```

```
input
9 3
1B5 158 2AF 662 2A F4 200 0 7C

output
0 2A 7C F4 158 1B5 200 2AF 662

explanation
The array after 1st digit is processed :   200 0 662 F4 1B5 158 2A 7C 2AF
The array after 2nd digit is processed :   200 0 2A 158 662 7C 2AF 1B5 F4
The array after 3rd digit is processed :   0 2A 7C F4 158 1B5 200 2AF 662
```

B: Bucket Sort

Bucket sort is a non-comparative sorting algorithm. It assumes that the data is drawn from a uniform distribution to achieve a average case time complexity of $O(n)$. For this question you will be working on floating point numbers and we want you to implement each bucket as a linked list and sort the list using only insertion sort just like how it was described in class .

Given an array of size n, you have to sort it using only bucket sort algorithm. You also have to print the number of elements in each bucket.

Input

The first line contains one positive integer n $(1 \leq n \leq 10^5)$, the size of array a.
The second line contains n numbers $a_1, a_2, \ldots, a_n$ $(0 \leq a_i < 1)$. (a_i will not have more than 6 digits)

Output

The first line of output should contain n numbers of the sorted array a.
The second line of output should contain n positive integers $b_0, b_1, \ldots, b_{n-1}$, the number of elements in each of the buckets. b_i holds values in the half-open interval $[i/n, (i+1)/n)$.

---

input
4
0.3 0.4 0.5 0.2

output
0.2 0.3 0.4 0.5
1 2 1 0

explanation
$\lfloor 0.3 \times 4 \rfloor = \lfloor 1.2 \rfloor = 1 \rightarrow 0.3$ goes into b_1
$\lfloor 0.4 \times 4 \rfloor = \lfloor 1.6 \rfloor = 1 \rightarrow 0.4$ goes into b_1
$\lfloor 0.5 \times 4 \rfloor = \lfloor 2.0 \rfloor = 2 \rightarrow 0.5$ goes into b_2
$\lfloor 0.2 \times 4 \rfloor = \lfloor 0.8 \rfloor = 0 \rightarrow 0.2$ goes into b_0
$\rightarrow b_0, b_1, \ldots, b_{n-1} = 1, 2, 1, 0$

---

input
9
0.8 0.5 0.2 0.6 0.2 0.1 0 0 0.7

output
0 0 0.1 0.2 0.2 0.5 0.6 0.7 0.8
3 2 0 0 1 1 1 1 0

C: Counting Sort

Counting sort is a non-comparative sorting algorithm. It assumes that the data contains integers in order of $O(k)$ to achieve time complexity of $O(n + k)$. We want you to implement stable counting sort where $k = n$. So, for this purpose you will need to sort the array and also print the index of each element (Assume 1-based indexing).

Given an array of size n, you have to sort it in time complexity $O(n)$ using only stable counting sort algorithm.

Input

The first line contains one positive integer n $(1 \leq n \leq 10^5)$, the size of array a.
The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq n)$.

Output

The first line of output should contain n integers of the sorted array a.
The second line of output should contain n integers - the index of the corresponding element.

---

input
4
3 4 4 2

output
2 3 4 4
4 1 2 3 3 2

---

input
2
2 1

output
1 2
2 1

---

input
9
8 5 2 6 2 1 0 0 7

output
0 0 1 2 2 5 6 7 8
7 8 6 3 5 2 4 9 1

4

D:AA Mex...

Given an array of all integers in the range 1 to n except for one integer (called missing integer), where all non-missing integers appear any number of times. Find the missing integer....

But before the question is completed, Nom ate more elements of the array, so Nom gives you an array of n integers in the range 1 to n containing some missing integers.. He wants you to find if there exist two missing integers whose sum is k.

Input

The first line contains two positive integers n k ($1 \leq n \leq 10^5$), the size of array a.
The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq n$).

Output

The only line of output should contain "yes"/"no" without quotes, the answer to the problem.

```
input
5 4
5 2 2 2 5

output
yes

explanation
1 is missing and 3 is missing, 1 + 3 = 4
Note:  We need a pair, so we can't take just 4
```

```
input
2 2
1 1

output
no
```

```
input
9 13
8 5 3 5 4 1 2 2 9

output
yes
```

E: Efficiently Sorting Pairs

A pair is a struct which holds 2 values. We refer to these values as the first and the second. For example consider a pair $p = (1, 2)$, here $p.first = 1$ and $p.second = 2$.

If you compare pairs based on only the first element, there might be pairs which cannot be compared. So for this reason we impose a total ordering on the pairs. $p < q$ if $p.first < q.first$ or if $p.first = q.first$ and $p.second < q.second$. Sorting an array of pairs means for every index i $p_i \leq p_{i+1}$ ($1 \leq i \leq n-1$), where p_i's are the elements of the array p.

Given an array of pairs of size n, you have to sort it in time complexity $O(n)$ using non-comparative based sorting (You need to solve this problem by creating a struct called pair having the members first and second as mentioned in the problem, other methods wont be tolerated).

Input

The first line contains one positive integer n ($1 \leq n \leq 10^5$), the size of array a.
The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq n$).
The third line contains n integers $b_1, b_2, \ldots, b_n$ ($1 \leq b_i \leq n$).
Here, $p_i = (a_i, b_i)$

Output

The first line of output should contain n integers $a_1, a_2, \ldots, a_n$.
The second line of output should contain n integers $b_1, b_2, \ldots, b_n$.
Where, $p_i = (a_i, b_i)$, where the array p is in sorted order.

```
input
4
3 4 4 2
1 2 3 4

output
2 3 4 4
4 1 2 3
```

```
input
9
8 5 2 6 2 1 0 0 7
2 4 8 2 6 1 4 9 3

output
0 0 1 2 2 5 6 7 8
4 9 1 6 8 4 2 3 2
```

F: Frequency Sort

Given an array of size n, sort the array in increasing order based on the frequency of the values. If multiple values have the same frequency maintain the same order as between the first occurrence of each value in the original array.

You have to do this in $O(n)$ using a non-comparative sorting algorithm.

Input

The first line contains one positive integer n $(1 \leq n \leq 10^5)$, the size of array a.
The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq n)$.

Output

The first line of output should contain n integers of the array a sorted as mentioned in the problem.

---

input
4
3 4 4 2

output
3 2 4 4

---

input
2
2 1

output
2 1

---

input
9
8 5 4 4 3 2 6 5 1

output
8 3 2 6 1 5 5 4 4

explanation
8,3,2,6,1 have the same frequency, but the order is being maintained.
5,4 have same frequency, but the order is being maintained.
First Occurrence of 5 = 2nd position,
First Occurrence of 4 = 3nd position

G: Ganyu Impact

Kira and Nom play Gemshim Impact all day. That's why they are single.

Nom is pro gamer, he knows Ganyu is the strongest character so that's why he wants to get her. But Kira is even more pro, he knows that since Ganyu is too strong she will make the game less challenging and thus less fun. But they both liked her story quest. Hardwork always pays off.

In her story quest, Ganyu was in middle of her adeptus test. In this test, there are n targets and Ganyu has to hit any k targets. Since she is a bow character, its easier for her to aim at her closest targets. Given Ganyu's position x, find positions of k closest targets to her, so that she passes her Adeptus Test (Closest to Farthest, in case of ties left most first).

Input

The first line contains three positive integer n, k and x ($1 \leq k \leq n \leq 10^5$, $-10^9 \leq x \leq 10^9$).
The second line contains n integers $a_1, a_2, \ldots, a_n$ ($-10^9 \leq a_i \leq 10^9$),
where a_i denotes position of i-th target (Assume 1-D, each value represents a x-coordinates).

Output

The only line of output should contain k integers - the positions of targets in the correct order.

---

input
4 3 3
3 4 5 2

output
3 2 4

---

input
2 1 1
6 4

output
4

---

input
9 5 4
8 5 2 6 2 1 0 0 7

output
5 2 2 6 1

H: HuTao Impact

Kira has so many 5* characters in Gemshim impact, he literally breathes in 5*.
One of Kira's favorite characters is HuTao, she is a very strong "Pyro" (aka fire) Character. Kira even got her very expensive weapon and now wants to put her in a good team.

HuTao works well with some characters (like Xinqui, Jongli) and has horrible synergy with some characters (like Jean, QiQi). Kira wants to maximize the Goodness of his HuTao team. Given n characters, their DPS and their synergy with HuTao and also the **max team size** k, determine the maximum possible Goodness of the team.

Goodness of a team is defined as the sum of DPS of the all characters in a team (excluding HuTao) plus the sum of synergy of the all characters in a team (excluding HuTao).

Input

The first line contains two positive integers n and k ($1 \leq k \leq n \leq 10^5$).
The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^9$), where a_i denotes DPS of i-th character.
The third line contains n integers $b_1, b_2, \ldots, b_n$ ($-10^9 \leq b_i \leq 10^9$), where b_i denotes synergy of i-th character.

Output

The only line of output should contain one integer, the Goodness of the optimal team.

```
input
4 2
3 4 5 2
1 -2 2 -10

output
11
```

```
input
9 6
3 5 8 2 4 2 3 7 1
-3 -6 -9 -3 -5 -4 3 -4 -2

output
9
```

can get less also

I: Impact de Gemshim

The reason why Kira has so many 5* is obvious, he spent a lot of money on Gemshim. But the sad thing is, he didn't get the character he wanted even after spending so much but just got some duplicates.

Kira is a changed person now, instead of spending money, he wants to make money. But Nom is about to start his addiction arc. He took Kira's character history and now wants to find which characters he should get.

Nom doesn't want to get a lot of duplicates since that would make him sad, he wanted to find the minimum possible sum of square of character count of any subset of k characters. More formally, given an array a containing lower-case characters, for any k-sized subset $S = (b_1, b_2, \ldots, b_k)$, find the minimum value of $\Sigma_{c=a}^{z} f^2(c)$, where $f(c) = \Sigma_{i=1}^{k}[b_i = c]$ (value of [...] is 1, if inside expression is true, otherwise its 0).

Input

The first line contains two positive integers n k ($1 \leq n \leq 10^5$), the number of characters of Kira and the number of characters Nom wants.
The second line contains n lower-case characters $a_1, a_2, \ldots, a_n$.

Output

The only line of output should contain one integer - the minimum sum.

```
input
5 4
a b c c c

output
6

explanation
We remove c to get the value as 1² + 1² + 2² = 6
```

```
input
4 2
a a a b

output
2
```

J: Jongli Impact

Nom is noob at Spiral Abyss. His characters keep dying, but this never happens to Kira. This is because Kira has a character called Jongli, who is the "Geo" Archon. He can manifest a shield which can protect a character against anything. Kira wants to shield his characters again.

Kira is smart and only wants to shield characters whose HP is less than x. Given his n - 5* characters, he wants to choose some k characters such that HP of every chosen character is less than x and also k is maximal. Find the indices of the chosen characters. (Assume 1-based indexing).

Input

The first line contains three positive integer n and x $(1 \leq n \leq 10^5, 1 \leq x \leq 10^9)$.
The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq 10^9)$, where a_i denotes HP of i-th character.

Output

The first line of output should contain one integer k, the number of characters in your solution. The second line of output should contain k integers denoting the indices of characters to consider.

```
input
4 4
3 4 5 2

output
2
1 4
```

```
input
9 6
3 5 8 2 4 2 3 7 1

output
7
9 4 6 1 7 5 2
```