Codeforces Round #311 (Div. 2)

A. Ilya and Diplomas

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Soon a school Olympiad in Informatics will be held in Berland, n schoolchildren will participate there.

At a meeting of the jury of the Olympiad it was decided that *each* of the n participants, depending on the results, will get a diploma of the first, second or third degree. Thus, each student will receive exactly one diploma.

They also decided that there must be given at least min_1 and at most max_1 diplomas of the first degree, at least min_2 and at most max_2 diplomas of the second degree, and at least min_3 and at most max_3 diplomas of the third degree.

After some discussion it was decided to choose from all the options of distributing diplomas satisfying these limitations the one that maximizes the number of participants who receive diplomas of the first degree. Of all these options they select the one which maximizes the number of the participants who receive diplomas of the second degree. If there are multiple of these options, they select the option that maximizes the number of diplomas of the third degree.

Choosing the best option of distributing certificates was entrusted to Ilya, one of the best programmers of Berland. However, he found more important things to do, so it is your task now to choose the best option of distributing of diplomas, based on the described limitations.

It is guaranteed that the described limitations are such that there is a way to choose such an option of distributing diplomas that all n participants of the Olympiad will receive a diploma of some degree.

Input

The first line of the input contains a single integer n ($3 \le n \le 3 \cdot 10^6$) — the number of schoolchildren who will participate in the Olympiad.

The next line of the input contains two integers min_1 and max_1 ($1 \le min_1 \le max_1 \le 10^6$) — the minimum and maximum limits on the number of diplomas of the first degree that can be distributed.

The third line of the input contains two integers min_2 and max_2 ($1 \le min_2 \le max_2 \le 10^6$) — the minimum and maximum limits on the number of diplomas of the second degree that can be distributed.

The next line of the input contains two integers min_3 and max_3 ($1 \le min_3 \le max_3 \le 10^6$) — the minimum and maximum limits on the number of diplomas of the third degree that can be distributed.

It is guaranteed that $min_1 + min_2 + min_3 \le n \le max_1 + max_2 + max_3$.

Output

In the first line of the output print three numbers, showing how many diplomas of the first, second and third degree will be given to students in the optimal variant of distributing diplomas.

The optimal variant of distributing diplomas is the one that maximizes the number of students who receive diplomas of the first degree. Of all the suitable options, the best one is the one which maximizes the number of participants who receive diplomas of the second degree. If there are several of these options, the best one is the one that maximizes the number of diplomas of the third degree.

Examples

input
6 1 5 2 6 3 7
output
1 2 3

input
10 1 2 1 3 1 5

output

```
2 3 5
```

input

```
6
1 3
2 2
2 2
```

output

```
2 2 2
```

B. Pasha and Tea

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Pasha decided to invite his friends to a tea party. For that occasion, he has a large teapot with the capacity of w milliliters and $2n$ tea cups, each cup is for one of Pasha's friends. The i-th cup can hold at most a_i milliliters of water.

It turned out that among Pasha's friends there are exactly n boys and exactly n girls and all of them are going to come to the tea party. To please everyone, Pasha decided to pour the water for the tea as follows:

- Pasha can boil the teapot exactly once by pouring there at most w milliliters of water;
- Pasha pours the same amount of water to each girl;
- Pasha pours the same amount of water to each boy;
- if each girl gets x milliliters of water, then each boy gets $2x$ milliliters of water.

In the other words, each boy should get two times more water than each girl does.

Pasha is very kind and polite, so he wants to maximize the total amount of the water that he pours to his friends. Your task is to help him and determine the optimum distribution of cups between Pasha's friends.

Input

The first line of the input contains two integers, n and w ($1 \le n \le 10^5$, $1 \le w \le 10^9$) — the number of Pasha's friends that are boys (equal to the number of Pasha's friends that are girls) and the capacity of Pasha's teapot in milliliters.

The second line of the input contains the sequence of integers a_i ($1 \le a_i \le 10^9$, $1 \le i \le 2n$) — the capacities of Pasha's tea cups in milliliters.

Output

Print a single real number — the maximum total amount of water in milliliters that Pasha can pour to his friends without violating the given conditions. Your answer will be considered correct if its absolute or relative error doesn't exceed 10^{-6}.

Examples

input
2 4 1 1 1 1
output
3

input
3 18 4 4 4 2 2 2
output
18

input
1 5 2 3
output
4.5

Note

Pasha also has candies that he is going to give to girls but that is another task...

C. Arthur and Table

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Arthur has bought a beautiful big table into his new flat. When he came home, Arthur noticed that the new table is unstable.

In total the table Arthur bought has n legs, the length of the i-th leg is l_i.

Arthur decided to make the table stable and remove some legs. For each of them Arthur determined number d_i — the amount of energy that he spends to remove the i-th leg.

A table with k legs is assumed to be stable if there are more than half legs of the maximum length. For example, to make a table with 5 legs stable, you need to make sure it has at least three (out of these five) legs of the maximum length. Also, a table with one leg is always stable and a table with two legs is stable if and only if they have the same lengths.

Your task is to help Arthur and count the minimum number of energy units Arthur should spend on making the table stable.

Input

The first line of the input contains integer n ($1 \le n \le 10^5$) — the initial number of legs in the table Arthur bought.

The second line of the input contains a sequence of n integers l_i ($1 \le l_i \le 10^5$), where l_i is equal to the length of the i-th leg of the table.

The third line of the input contains a sequence of n integers d_i ($1 \le d_i \le 200$), where d_i is the number of energy units that Arthur spends on removing the i-th leg off the table.

Output

Print a single integer — the minimum number of energy units that Arthur needs to spend in order to make the table stable.

Examples

input
2 1 5 3 2
output
2

input
3 2 4 4 1 1 1
output
0

input
6 2 2 1 1 3 3 4 3 5 5 2 1
output
8

D. Vitaly and Cycle

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

After Vitaly was expelled from the university, he became interested in the graph theory.

Vitaly especially liked the cycles of an odd length in which each vertex occurs at most once.

Vitaly was wondering how to solve the following problem. You are given an undirected graph consisting of n vertices and m edges, not necessarily connected, without parallel edges and loops. You need to find t — the minimum number of edges that must be added to the given graph in order to form a simple cycle of an odd length, consisting of more than one vertex. Moreover, he must find w — the number of ways to add t edges in order to form a cycle of an odd length (consisting of more than one vertex). It is prohibited to add loops or parallel edges.

Two ways to add edges to the graph are considered equal if they have the same sets of added edges.

Since Vitaly does not study at the university, he asked you to help him with this task.

Input

The first line of the input contains two integers n and m ($3 \le n \le 10^5, 0 \le m \le \min(\frac{n(n-1)}{2}, 10^5)$) — the number of vertices in the graph and the number of edges in the graph.

Next m lines contain the descriptions of the edges of the graph, one edge per line. Each edge is given by a pair of integers a_i, b_i ($1 \le a_i, b_i \le n$) — the vertices that are connected by the i-th edge. All numbers in the lines are separated by a single space.

It is guaranteed that the given graph doesn't contain any loops and parallel edges. The graph isn't necessarily connected.

Output

Print in the first line of the output two space-separated integers t and w — the minimum number of edges that should be added to the graph to form a simple cycle of an odd length consisting of more than one vertex where each vertex occurs at most once, and the number of ways to do this.

Examples

input
4 4 1 2 1 3 4 2 4 3
output
1 2

input
3 3 1 2 2 3 3 1
output
0 1

input
3 0
output
3 1

Note

The simple cycle is a cycle that doesn't contain any vertex twice.

E. Ann and Half-Palindrome

time limit per test: 1.5 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Tomorrow Ann takes the hardest exam of programming where she should get an excellent mark.

On the last theoretical class the teacher introduced the notion of a *half-palindrome*.

String t is a *half-palindrome*, if for all the odd positions i ($1 \le i \le \frac{|t|+1}{2}$) the following condition is held: $t_i = t_{|t|-i+1}$, where $|t|$ is the length of string t if positions are indexed from 1. For example, strings "abaa", "a", "bb", "abbbaa" are half-palindromes and strings "ab", "bba" and "aaabaa" are not.

Ann knows that on the exam she will get string s, consisting only of letters a and b, and number k. To get an excellent mark she has to find the k-th in the lexicographical order string among all substrings of s that are half-palyndromes. Note that each substring in this order is considered as many times as many times it occurs in s.

The teachers guarantees that the given number k doesn't exceed the number of substrings of the given string that are half-palindromes.

Can you cope with this problem?

Input

The first line of the input contains string s ($1 \le |s| \le 5000$), consisting only of characters 'a' and 'b', where $|s|$ is the length of string s.

The second line contains a positive integer k — the lexicographical number of the requested string among all the half-palindrome substrings of the given string s. The strings are numbered starting from one.

It is guaranteed that number k doesn't exceed the number of substrings of the given string that are half-palindromes.

Output

Print a substring of the given string that is the k-th in the lexicographical order of all substrings of the given string that are half-palindromes.

Examples

input
abbabaab 7
output
abaa

input
aaaaa 10
output
aaa

input
bbaabb 13
output
bbaabb

Note

By definition, string $a = a_1a_2\ldots a_n$ is lexicographically less than string $b = b_1b_2\ldots b_m$, if either a is a prefix of b and doesn't coincide with b, or there exists such i, that $a_1 = b_1, a_2 = b_2, \ldots a_{i-1} = b_{i-1}, a_i < b_i$.

In the first sample half-palindrome substrings are the following strings — a, a, a, a, aa, aba, abaa, abba, abbabaa, b, b, b, b, baab, bab, bb, bbab, bbabaab (the list is given in the lexicographical order).