CS F211
Data Structures and Algorithms
Assignment - 3

Allowed languages: **C**

February 3, 2021

General Tips

- 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. You can install Windows Subsystem Linux (WSL) or MinGW 7.3.0, if you are Windows user to compile and run your programs. Alternatively, you can run and test your codes on Online GDB. If you are using WSL or Linux to run your programs, make sure that the gcc version is `gcc 5.4.1 c99`.

A: Ice Cream

There are N kids and M ice cream cones. The i^{th} ice cream cone has a size S_i. Each kid has a preferred cone size with the i^{th} child preferring a size A_i. Each child will accept an ice cream cone if the size of the ice cream cone is between $A_i \pm k$ inclusive. You now have to find the largest number of children that will get an ice cream cone if you distribute the ice cream cones optimally. *Please note that you cannot give an ice cream cone to more than one kid and each child can have no more than one ice cream cone.*

Input

The first line of input contains three space-separated integers N and M $(1 \leq N, M \leq 10^5)$ and $(1 \leq k \leq 10^9)$. The second line contains N space-separated integers representing the array A that is the preferred sizes of ice cream cones for each of the N kids. The third line contains M space-separated integers representing the size of the array S, the sizes of the available ice cream cones.

Output

The output should have exactly one integer, the maximum number of children that can get an ice cream cone if they are distributed optimally.

---

input
```
4 3 5
60 45 60 80
30 75 60
```

output
```
2
```

explanation

here a kid with preference 60 can get the ice cream cone with size 60 and the kid with preference 80 can get an ice cream cone of size 75. Hence the answer is 2. It is easy to see that we can satisfy no more than two kids.

B: Assassins

You have a total of N assassins and each has a skill a_i. There also exist M nobles. Each of these nobles has a bodyguard with a skill b_i and a certain amount of gold g_i. An assassin can kill a bodyguard of a noble if the assassin's skill is greater than or equal to the bodyguard's skill ($a_i \geq b_j$). If an assassin kills a bodyguard he can steal all the gold of the noble. How much gold can each assassin steal? Please calculate the answer for each assassin *independent* of the other assassins. Do not assume that if a noble's gold is stolen by one assassin then other assassins can't steal from him. The assassins are not actually killing the body guards as such, you just need to find how much gold each of them *can* steal hypothetically.

Input

The first line contains two integers N and M ($1 \leq N, M \leq 10^5$), the number of assassins and nobles. The second line contains N integers representing the array a, where a_i is the skill of the i^{th} assassin ($-10^9 \leq a_i \leq 10^9$). Then M lines follow, where the i^{th} line contains the two integers b_i and g_i, the skill of the bodyguard of the i^{th} noble and the amount of his gold. ($-10^9 \leq b_i, g_i \leq 10^9$)

Output

Print one line containing N integers, where the i^{th} integer represents the maximum gold that the i^{th} assassin can steal.

---

input

```
5 4
1 4 3 2 5
4 2
0 1
2 8
9 4
```

output

```
1 11 9 9 11
```

explanation

```
The first assassin can only steal gold from the second noble.  The second can steal
gold from the first, second, and third nobles.  The third can steal from the second
and third.  The fourth can steal from second and third.  The fifth can steal from first,
second and third.
```

C: Tree Planting

You want to plant N trees in your garden. Your garden can be represented by a number line that contains fertile spots at certain points. In particular there are M fertile spots ($M \geq N$), x_1, x_2, x_3 ... x_M where you can plant a tree. You can only plant a tree in a fertile spot and a fertile spot can have a maximum of one tree. As we know from our high school biology, two trees cannot be kept too close to each other otherwise they will take up each others water and nutrients. To make sure that all the trees are healthy, you want to plant them in such a way that the minimum distance between any two of them is as large as possible. What is the largest possible minimum distance between any two trees?

Input

The first line of the input contains two space-separated integers N and M, ($1 \leq N \leq M \leq 10^5$). The second line of input contains M space-separated integers, x_1, x_2, x_3 ... x_M, representing the co-ordinates of the fertile spots ($0 \leq x_i \leq 10^9$).

Output

Output a single integer which is the largest possible minimum distance you can get by planting the trees in some way.

```
input
3 5
2 1 8 4 9

output
3

explanation
we can get a minimum distance of 3 if we place trees at positions 1,4 and 8.  It can
be easily seen that it is not possible to get a larger minimum distance no matter where
you plant the trees
```

D: Good Pairs

Given two arrays x and y, both containing N elements, find the number of pairs of integers i, j such that $x_i + x_j + k_1 > y_i + y_j + k_2$ where $(i < j)$.

Input

The first line contains three integers N, the size of the arrays, and k_1 and k_2 ($1 \leq N \leq 10^5$, $-10^9 \leq k_1, k_2 \leq 10^9$). The second line contains N integers representing the array a. The third line contains N integers representing the array b ($-10^9 \leq a_i, b_i \leq 10^9$).

Output

Print a single integer, the number of such pairs.

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

output
7

explanation
```

The pairs i, j are $(1,4),(2,4),(3,4),(4,5),(1,2),(2,3),(2,5)$. We can easily verify that no other pair satisfies this inequality

---

E: The Simplified Logo Compiler

Logo is a programming language that can be used to draw simple shapes on the screen. In this question, you will have to implement a simplified Logo compiler that supports to commands 'FD' and 'LOOP...END' statements for a one dimensional turtle. The program begins with the turtle (cursor) located at coordinate 0, and the command "FD x" (x is an integer) can be used to move the turtle by x units. Loop instructions consist of a line beginning with "LOOP M" (M is an integer) and end with the line "END". The commands between the LOOP and END need to be repeated M times. Given a valid logo program with these two commands, provide the final location of the turtle.

Input

The first line consists of one integer N ($N \leq 10^4$), the number of lines in the logo program. The next N lines describe the program, and each line will have a maximum of 32 characters.

Output

Print one integer X, the final position of the turtle.

```
input
2
FD 40
FD -30

output
10
```

```
input
5
FD 50
LOOP 3
FD 10
FD 25
END

output
155
```

```
input
9
FD 40
LOOP 10
FD 5
LOOP 7
FD -5
FD 7
END
FD 6
END

output
290
```

explanation

$40 + 10 \times \{5 + 7 * (-5 + 7) + 6\}$

F: Fighting Fire With Fire

Moontech Pharmacueticals has successfully created an *anti-virus virus* that can be used to fight COVID-19. They intend to start injecting the new virus into people as soon as possible, but the astronomical cost of each dose means they want to *minimize* the number of total doses needed. The *antivirus* works like a regular virus, and can spread from one human to another, and is highly contagious. Given a population of N people, and a list of M *friendships* (people who will spend enough time with each other for the antivirus to spread), find the minimum number of people who need to be vaccinated to reach herd immunity (defined to be strictly greater than 80% of the population). Note that if A is a friend of B and B is a friend of C, injecting A with the antivirus will ensure that C also gets infected.

Input

The first line of input contains the integer N (population) and M (number of friendships) ($1 \leq N, M \leq 10^5$, and individuals of the population are numbered $0 \ldots N-1$. The next M lines contain two integers u, v representing that u is friends with v and vice versa.

Output

Print one integer, E, denoting the number of antivirus doses needed to infect strictly greater than 80% of the population.

---

input

```
10 8
0 1
1 8
5 7
8 0
6 9
6 7
9 5
2 3
```

output

```
3
```

explanation

```
Infecting any one of (0, 1, 8) will ensure all three of them are infected.  Similarly
for the sets (2, 3) and (5,6,7,9).  For example, we can infect 5, 1 and 2 to reach
a total of 9 people infected, which is strictly greater than 80%.
```

G: Hitchcock and Scully

Hitchcock and Scully are trying to find new places to visit for lunch over the next N days. Since they have only a limited amount of time in their lunch break, they make a list of M restaurants in Brooklyn's 99^{th} precinct. Each restaurant will cost them $\$m_i$ for a lunch. Given a list of size N, representing the (total) amount of money they have in their pockets for each of the next N days, calculate how many *options* they have for lunch each day. They can go to any restaurant they want on day j, provided $n_j \leq m_i$.

Input

The first line of input contains space-separated M $(1 \leq M \leq 10^5)$ and N $(1 \leq N \leq 10^5)$. The second line contains M space separated numbers m_i, representing the cost of dining at restaurant i. The third line contains N space separated integers representing the money Hitchcock and Scully have each day. It is guaranteed that $1 \leq m_i, n_j \leq 10^9$.

Output

Print N space separated integers, representing the number of choices they have for each day.

```
input
10 5
8 9 6 5 4 3 23 9 10 1
24 9 12 3 1

output
10 8 9 2 1

explanation
On the first day with $24, they can
visit all ten restaurants.  On day
2, they can visit all restaurants
except the one costing $24.
```

```
input
5 3
60 40 90 45 120
13 44 90

output
0 1 4

explanation
If they have no options for a
certain day, print 0.
```

H: H-Index

Given a graph of publications and citations, with each node representing a publication and each edge representing **one** citation, calculate the H-index of all the authors. We have A authors (numbered $0 \ldots A-1$) and P publications (numbered $0 \ldots P-1$). The value of h-index (h) of an author is the maximal possible value x, such that the number of papers (x) by the author have x or more citations each. Each publication is written by exactly one author. There are a total of C citations (edges) in the academic graph.

Input

The first line of input contains the numbers A, P, C in a space separated fashion such that $0 \leq A \leq P \leq 1000$. The second line contains P numbers - the author for each of the publications. The next C lines contain two space separated values indicating the publication numbers (p_i, p_j) meaning that p_i cited p_j.

Output

Print A space separated integers denoting the H-index of all of the authors.

---

input

```
3 5 7
0 0 1 1 2
1 0
2 0
3 0
4 0
2 1
3 1
1 4
```

output

```
2 0 1
```

explanation

```
We have three authors and five publications.  The number of citations for each publication
is:  4 2 0 0 1.  Author 0 has 2 publications with 4, 2 citations respectively.  So,
his H-index is 2.
```

I: Okabe and the Toll Gates

The cities in Japan lie on a straight line numbered from 0 serially and adjacent cities are unit distance apart. Okabe Rintaro lives in city U has been invited to give a talk on his Time Machine theory in city V. He plans to rent a car from his city to the destination. The road from city U to V has a few toll gates. Each toll gate has a gas station. All the gas stations surprisingly sell gas in fixed capacity containers (in litres). A litre of gas costs K yen. The car Okabe rents runs according to the following mileage: Z litres of gas lets him drive $AZ + B$ units. On his drive, Okabe plans to stop at every immediate toll gate (not anywhere in between), empty any gas currently in the tank and refill it again from the gas station at the toll gate. Help Okabe spend as minimum money as possible on the gas. It is guaranteed that cities U and V will always have tolls gates. Assume that the car had no gas before Okabe rented it. *Note that, the use of inbuilt qsort function cannot be made to solve this problem.*

Input

The first line of input contains three space-separated N ($1 \leq$ N $\leq 10^5$), M ($1 \leq$ M $\leq$ N) and L ($1 \leq$ L $\leq 10^5$) denoting the number of cities in Japan, the number of cities that have toll gates and the number of different gas containers sold at each gas station. The next line contains five space-separated integers U ($0 \leq$ U $\leq$ N - 1), V (U $\leq$ V $\leq$ N - 1), A, B ($1 \leq$ A, B $\leq 10^3$) and K ($1 \leq$ K $\leq 10^3$), denoting the starting and destination city, mileage coefficients and the rate of gas respectively. The following line has M space-separated integers ($0 \leq T_i \leq$ N - 1) denoting the cities that have toll gates. The last line of input contains L space-separated integers ($1 \leq C_i \leq 10^6$) denoting the various gas container sizes sold in the gas stations.

Output

Print a single integer P, denoting the minimum money Okabe should spend on gas. If Okabe cannot make it to the city V using the above strategy, print "NOT POSSIBLE".

```
input
100 15 7
11 92 2 3 10
52 81 76 36 5 23 50 90 17 46 3 82 11 92 83
1 8 11 7 2 5 9

output
330
```

J. Okabe and Entropy

Okabe, having completed the course on Advanced Statistical Mechanics in his university, has an epiphany as to how to solve the problem of the parallel worldlines. He realizes that to transit from one worldline to another, he needs exactly E_i energy (transit potential) to overcome the entropy between those two worldlines. After painstaking calculations, Okabe has finally figured out all the transit potentials. But to solve the final problem, he needs one extra information, that is the cost of the Minimum Spanning Tree across these worldlines. As he is dog-tired, he turns to you, Makise Kurisu, to help him find that cost. *Note: You can read up more about MSTs and how to find their cost here.*

Input

The first line of input contains two space-separated integers N ($2 \leq$ N ≤ 500), M ($1 \leq$ M $\leq \frac{N(N-1)}{2}$), denoting the number of worldlines and the number of transits possible between those worldines respectively. The following M lines contain three space-separated integers U_i, V_i and E_i ($0 \leq U_i$, $V_i \leq$ N-1, $1 \leq E_i \leq 10^5$) denoting an undirected transit between the worldlines U_i and V_i which has a transit potential of E_i. It is guaranteed that worldlines graph will be connected.

Output

Print a single integer E, denoting the cost of the Minimum Spanning Tree of the worldlines graph.

```
input
7 10
0 1 1
0 2 5
2 3 1
3 4 3
4 5 4
5 0 3
6 1 3
2 6 2
3 6 2
6 4 3
6 5 2
0 6 2

output
11
```
