

2021 網際網路程式設計全國大賽

高中組網路賽

- 本次比賽共 9 題，含本封面共 26 頁。
- 全部題目的輸入都來自**標準輸入**。輸入中可能包含多組輸入，以題目敘述為主。
- 全部題目的輸出皆輸出到螢幕（**標準輸出**）。
輸出和裁判的答案必須完全一致，英文字母大小寫不同或有多餘字元皆視為答題錯誤。
- 比賽中上傳之程式碼，使用 C 語言請用 `.c` 為副檔名；使用 C++ 語言則用 `.cpp` 為副檔名。
- 使用 `cin` 輸入速度遠慢於 `scanf` 輸入，若使用需自行承擔 Time Limit Exceeded 的風險。
- 任何題目內提到的「一行」，皆代表以換行字元「`\n`」結尾的字串。
- 部分題目有浮點數輸出，會採容許部分誤差的方式進行評測。一般來說「相對或絕對誤差不超過 ϵ 皆視為正確」， ϵ 值以題目敘述為主。

舉例來說，假設 $\epsilon = 10^{-6}$ 且 a 是正確答案， b 是你的答案，如果符合 $\frac{|a-b|}{\max(|a|, |b|, 1)} \leq 10^{-6}$ ，就會被評測程式視為正確。

Problem	Problem Name	Time Limit	Memory Limit
A	殿王的飲料王國	1 s	1024 MB
B	小 P 打仗	2 s	1024 MB
C	箭頭謎題	1 s	1024 MB
D	紗霧與正宗	6 s	1024 MB
E	計算機結構	12 s	1024 MB
F	隨機排序	3 s	1024 MB
G	陣列刪除	1 s	1024 MB
H	殿王愛字串	2 s	1024 MB
I	幸福感	3 s	1024 MB

2021 網際網路程式設計全國大賽

輸入輸出範例

C 程式範例：

```
1 #include <stdio.h>
2 int main()
3 {
4     int cases;
5     scanf("%d", &cases);
6     for (int i = 0; i < cases; ++i)
7     {
8         long long a, b;
9         scanf("%lld %lld", &a, &b);
10        printf("%lld\n", a + b);
11    }
12    return 0;
13 }
```

C++ 程式範例：

```
1 #include <iostream>
2 int main()
3 {
4     int cases;
5     std::cin >> cases;
6     for (int i = 0; i < cases; ++i)
7     {
8         long long a, b;
9         std::cin >> a >> b;
10        std::cout << a + b << std::endl;
11    }
12    return 0;
13 }
```

A. 殿王的飲料王國

Problem ID: drinks

殿王是個天才兒童，他在一個月大的時候就學會數數、六個月大的時候就學會乘法跟除法、一歲時學會寫程式、一歲又六個月時養了可愛的拉布拉多、一歲又十個月時養了可愛的貓咪、兩歲時發明了「吃餅乾」的遊戲、三歲又三個月大時成功地對貓咪做了排序。現在要講的是殿王三歲又四個月的故事。

殿王在三歲又四個月的時候創立了一個飲料王國，這個王國一共有 N 種面額的貨幣，面額大小分別為 $1, 2, \dots, N$ 元。某一天，王國中的 $N \times M$ 位小朋友想一起去飲料王國中最有名的飲料店「億園」買飲料，「億園」之所以有名是因為他販售了史上最好喝的飲料「QQ捏捏好喝到殿王茶」，而且只需要花 1 元就能購買。這 $N \times M$ 位小朋友中，有 M 位恰有一張面額為 1 元的貨幣、 M 位小朋友恰有一張面額為 2 元的貨幣、 \dots 、 M 位小朋友恰有一張面額為 N 元的貨幣。這 $N \times M$ 位小朋友都想去買「QQ捏捏好喝到殿王茶」，然而因為人數實在太多，殿王只好將小朋友們分成紅白兩隊，讓紅隊的人先去買飲料，等紅隊的所有人買完再讓白隊的人去買飲料。已知殿王在幫小朋友們分完隊伍後，紅白兩隊當中，分別有 a_i, b_i 位小朋友持有 i 元貨幣。

身為「億園」飲料店唯一店員的你，每次只能接受一張訂單，將手上唯一一張訂單處理完後才能受理下一位客人點餐。然而店面開張當天，你發現了一個大問題，那就是收銀機完全沒有任何的零錢，也就是沒有辦法找零。無奈之下你只能使用客人們支付的零錢進行找零，於是你希望小朋友們能以「完美的」順序排隊，使得每一位小朋友在點餐時，你都有足夠的零錢可以支付找零，也就是說，如果一位小朋友支付了面額為 x 元的貨幣，你必須能夠恰好找給他 $x - 1$ 元。

好奇心旺盛的你，想知道小朋友們有幾種「完美的」排隊順序可以讓你順利的為每位小朋友進行找零。在兩種排隊順序中，只要存在正整數 k 使得你在這兩個排隊順序中服務的第 k 位小朋友是不同人，那它們就是不相同的排隊順序。因為答案可能很大，所以只需要計算其除以 998244353 之後的餘數就可以了。

Input

輸入第一行包含兩個正整數 N, M 。

接下來共有 N 行，其中第 i 行包含兩個整數 a_i, b_i ，分別代表紅隊與白隊中持有 i 元貨幣的小朋友人數。

- $1 \leq N \times M \leq 10^6$
- $0 \leq a_i, b_i \leq M$
- $a_i + b_i = M$

Output

請輸出一個整數，代表一共有幾種「完美的」排隊順序除以 998244353 後的餘數。

Notes

在 Sample Input 1 中，假設 x_1, x_2 是兩位恰有一張面額為 1 元的小朋友， y_1, y_2 是兩位恰有一張面額為 2 元的小朋友，並且 $\{x_1, x_2, y_1\}$ 隸屬於紅隊， $\{y_2\}$ 隸屬於白隊，那麼所有完美的排隊順序如下：

- x_1, x_2, y_1, y_2
- x_2, x_1, y_1, y_2
- x_1, y_1, x_2, y_2
- x_2, y_1, x_1, y_2

Sample Input 1	Sample Output 1
2 2 2 0 1 1	4

Sample Input 2	Sample Output 2
3 2 2 0 2 0 1 1	20

B. 小 P 打仗

Problem ID: war

在小 P 國裡，有 N 個士兵隨時防守著城牆。由於疫情險峻，所以小 P 決定要替每位士兵施打疫苗。已知第 i 位士兵的防禦力為 s_i ，且在施打疫苗後，必須休息 r_i 天。也就是說，如果第 i 位士兵在第 D 天施打疫苗，那在第 $D, D+1, \dots, D+r_i-1$ 天，都必須休息。

讓我們定義 $w(i)$ 為所有在第 i 天不必休息的士兵的防禦力和。

為了避免新冠肺炎擴散，小 P 在接下來的 N 天中，每天必須**恰好為一位**士兵施打疫苗，且每位士兵都必須被施打**恰一劑**疫苗。

因為敵人每天都有可能攻打過來，所以小 P 想問你，如果敵人在第 k 天攻打過來，那 $w(k)$ 最大可以是多少？

請對於 $k = 1, 2, 3, \dots, N$ 都輸出一個整數代表答案。注意對於每個 k ，你可以獨立的決定對於這個 k 的施打順序。

Input

輸入的第一行有一個正整數 N ，代表有幾名士兵。

第二行有 N 個以空格分開的整數 s_1, s_2, \dots, s_N ，代表第 i 個士兵的防禦力為 s_i 。第三行有 N 個以空格分開的整數 r_1, r_2, \dots, r_N ，代表第 i 個士兵打完疫苗要休息 r_i 天。

- $1 \leq N \leq 3 \times 10^5$
- $1 \leq s_i \leq 10^9$
- $1 \leq r_i \leq N$

Output

請輸出 N 個以空白隔開的整數，第 i 個整數代表最大的 $w(i)$ 。請注意不要輸出多餘的行尾空白。

Sample Input 1

5 1 1 1 1 1 1 5 5 4 4	Sample Output 1 4 4 3 2 2
-----------------------------	-------------------------------------

Sample Input 2

10 20 9 1 38 2 1 3 4 8 23 1 3 3 2 9 9 8 9 10 10	Sample Output 2 108 108 108 108 108 106 103 99 94 75
---	--

C. 箭頭謎題

Problem ID: arrowpuzzle

OET 是一位喜愛並且精通放置遊戲的玩家。各式各樣的放置遊戲帶給他快樂。在遊玩「對數忙碌」的時候，他遇上了一個小遊戲：箭頭謎題。

在箭頭謎題這個小遊戲中，有 N 個箭頭由左至右排成一列。每個箭頭一開始可能會是向上或是向下的，每當你點擊一個箭頭，該箭頭以及與其相鄰的所有箭頭的狀態都會被反轉，也就是說，如果某個箭頭是向下的，那麼直接點擊該箭頭或是點擊任何與其相鄰的箭頭都會使得它變為向上，反之亦然。遊戲目標是在經過一系列的點擊操作之後，使得所有箭頭都是向上的，當你達成這個目標就可以得到一定的「星星」，是遊戲中的關鍵貨幣。

OET 是追求效率的玩家，每有空閒就會遊玩箭頭謎題賺取「星星」。他發現，有時候怎麼玩都不可能讓所有箭頭都是向上的狀態，只能按「我放棄！」按鈕重新來過。

現在告訴你所有箭頭的初始狀態（向上或是向下），請你幫 OET 快速辨別他是否有可能完成這局遊戲，也就是讓所有箭頭都是向上的狀態。

Input

第一行有一個正整數 N ，表示箭頭的數量。

第二行有一個長度 N 的 01 字串 S 表示所有箭頭的初始狀態， $S_i = 0$ 表示第 i 個箭頭的狀態是向上的， $S_i = 1$ 則表示第 i 個箭頭的狀態是向下的。

- $1 \leq N \leq 3 \times 10^5$
- $|S| = N, S_i \in \{0, 1\}$

Output

請輸出 Yes 或 No，代表 OET 是否有可能完成這局遊戲，也就是讓所有箭頭都是向上的狀態。

Sample Input 1	Sample Output 1
4 0101	Yes

Sample Input 2	Sample Output 2
2 01	No

Sample Input 3	Sample Output 3
7 1011001	Yes

D. 紗霧與正宗

Problem ID: sagiri

紗霧與正宗是一對感情非常好的兄妹。他們除了會一起畫漫畫、做家事、看煙火之外，還會一起玩撲克牌。

他們玩撲克牌的方式很特別，跟一般人的撲克牌玩法不太一樣。

首先，紗霧手上會先拿著 N 張撲克牌，對於第 i 張撲克牌，紗霧會給他一個數值 a_i ，代表紗霧對於那張撲克牌的喜好程度。撲克牌的喜好程度可以是正的（代表紗霧喜歡那張撲克牌），也可以是負的（代表紗霧討厭那張撲克牌），也可以是零（代表紗霧對那張撲克牌沒感覺）。

接著，正宗會與紗霧進行 K 輪的交易。第 i 輪交易的進行方式如下：

- 正宗會給紗霧一張喜好程度為 b_i 的牌。
- 紗霧必須丟出一張牌。她可以從以下兩個操作中選擇一個：
 - 直接把正宗給她的牌丟掉。
 - 把她手上的某一張牌丟掉，並且把正宗給她的牌放進那張被丟掉的牌的位置。
- 正宗會把紗霧丟掉的牌放進家裡的垃圾桶。

進行完 K 輪交易，他們會把那 N 張撲克牌按照原本的順序，由左至右的放在地板上。接著，正宗與紗霧會個別拿起一枝筆，指向撲克牌中的兩個位置（指向的位置可以一樣），並且把介於那兩個位置中間的撲克牌通通拿出來（包含那兩張被筆指的撲克牌），放進他們的作品之中。

而正宗與紗霧很好奇，他們所有可能的作品之中的撲克牌喜好程度總和，最大可以到多大？

Input

輸入的第一行包含一個正整數 T ，代表接下來測試資料的筆數。

每一筆測試資料，總共會占三行。

第一行包含兩個正整數 N, K ，代表，一開始紗霧手上的撲克牌數量，以及進行交易的次數。

第二行包含 N 個正整數 a_1, a_2, \dots, a_N ， a_i 代表紗霧手上第 i 張撲克牌的喜好程度。

第三行包含 K 個正整數 b_1, b_2, \dots, b_K ， b_i 代表正宗在第 i 輪交易時，會拿出來的撲克牌的喜好程度。

- $1 \leq T \leq 10^6$
- $1 \leq N \leq 10^6$
- $1 \leq K \leq 10^6$
- $-10^9 \leq a_i, b_i \leq 10^9$
- 在這 T 筆測試資料中， N 的總和不會超過 10^6
- 在這 T 筆測試資料中， K 的總和不會超過 10^6

Output

對於每一筆測試資料，請輸出一行，代表所有可能作品中的喜好程度總和的最大值。

Notes

在範例的第一組測試資料中，紗霧可以在第一輪交易時，把第三張牌換掉。這時候她手上的牌的喜好程度分別就是 $[-1, 3, -2, 3]$ 。如果他們指向第二張牌、第四張牌的話，可以得到喜好程度總和 4，這也是所有喜好程度總和的可能值中，最大的那一個。

在範例的第二組測試資料中，紗霧在第一輪交易時，可以直接把正宗給他的牌丟掉。接著，他們可以指向第一張牌、第五張牌，就可以得到喜好程度總和 15，這也是所有喜好程度總和的可能值中，最大的那一個。

Sample Input 1	Sample Output 1
2 4 1 -1 3 -6 3 -2 5 1 1 2 3 4 5 -1	4 15

This page is intentionally left blank.

E. 計算機結構

Problem ID: architecture

小 Y 是個熱愛學習的資工系學生，為了確保自己能夠更有效率地複習，小 Y 習慣從教授的投影片中擷取精華，製作自己的筆記。

某一天計算機結構（Computer Architecture，臺大資工系大三必修課程）下課後，小 Y 一如往常的打開這天上課的投影片打算開始製作筆記。這天的投影片上有一個重要的圖形，所以小 Y 打算將它也畫到自己的筆記本上。不過小 Y 很快地就遇到了問題：由於投影片上教授用了許多種顏色來代表圖形中各個不同意義的部分，小 Y 也想在自己筆記本上使用相對應的顏色。可是這個圖形實在太複雜，使得小 Y 不知道要用什麼順序跟方法來畫它！

投影片上的圖形可以被視為一個 N 個點以及 M 條邊的簡單無向圖，每個邊上都有編號為 1 到 K 的其中一種顏色。小 Y 也有這 K 種顏色的色筆。一開始，小 Y 的筆記本上已經畫好了這 N 個點，由於小 Y 是完美主義者，他每次都想要挑一個顏色，並使用這個顏色的色筆畫一個簡單環。如果著色的過程中經過了一條已經被塗色過的邊，那麼舊的顏色會被新的顏色蓋過去。

小 Y 不確定在這種只畫簡單環的方式下，能不能成功的畫出投影片上的圖片。請你寫一個程式幫他判斷吧！

一個簡單環被定義為一串長度大於 3 的序列 v_1, v_2, \dots, v_c ，滿足 $v_1 = v_c$ 、 v_1, v_2, \dots, v_{c-1} 兩兩相異且對於所有的 $1 \leq i < c$ ，都存在連接 v_i 和 v_{i+1} 的邊。

Input

輸入的第一行為一個正整數 T ，代表測試資料的數量。

每一筆測試資料的第一行有三個整數 N, M, K ，分別代表點的數量、邊的數量、以及顏色的數量。接著 M 行，第 i 行有三個正整數 u_i, v_i, c_i ，分別代表第 i 條邊是從第 u_i 個點連到第 v_i 個點，且顏色為 c_i 。

- $1 \leq T \leq 100$
- $1 \leq N \leq 2000$
- $0 \leq M \leq \min(\frac{N \times (N-1)}{2}, 10^6)$
- $1 \leq K \leq 2000$

- $1 \leq u_i \neq v_i \leq N$
- $1 \leq c_i \leq K$
- 保證輸入沒有重複的邊
- 保證 T 筆測試資料中 N 的總和不超過 2000
- 保證 T 筆測試資料中 M 的總和不超過 10^6
- 保證 T 筆測試資料中 K 的總和不超過 2000

Output

輸出 T 行，每行代表一筆測試資料的答案：如果小 Y 可以成功畫出這個圖形，輸出 Yes，否則，輸出 No。

Notes

對於第三筆範例輸入，一種合法的畫法是先將環 $2 - 3 - 6 - 5 - 2$ 塗顏色 2，接著再將環 $1 - 2 - 5 - 4 - 1$ 塗為顏色 1。

Sample Input 1	Sample Output 1
3 3 3 1 1 2 1 2 3 1 3 1 1 4 4 2 1 2 1 2 3 1 3 1 1 1 4 2 6 7 2 1 2 1 2 3 2 1 4 1 2 5 1 3 6 2 4 5 1 5 6 2	Yes No Yes

This page is intentionally left blank.

F. 隨機排序

Problem ID: randomshuffle

身經百戰、參加了各式各樣的程式競賽的小 Y，今天遇到了一個與眾不同的題目如下：

「有一個 $1 \sim N$ 的祕密排列 $P = [p_1, p_2, \dots, p_N]$ ，但你不知道 P 確切的排列內容。現在有 M 個線索，每個線索長得形如 $a_i < b_i$ 或是 $a_i > b_i$ ，代表 $p_{a_i} < p_{b_i}$ 或是 $p_{a_i} > p_{b_i}$ ，請你輸出 P 到底長什麼樣子。」

小 Y 覺得這題實在太簡單了，瞬間寫完 AC code 並且上傳！然而他得到的結果是 WA (Wrong Answer)。小 Y 與他的隊友都覺得很奇怪，小 Y 怎麼可能寫出會 WA 的 code 呢？

經過 0.1 秒的檢查，小 Y 發現這題並沒有說清楚在 P 有多組解的時候要輸出字典序最小或是最大的 P ，或者是輸出任一組解。在等待發問被回答的期間，他決定使用隨機大法來嘗試直接 AC 這題。更精確的說，他會在所有符合每一個線索的排列當中，均勻隨機的選擇一個輸出。小 Y 的隊友，小 P，也是一位通靈大師，已經猜到主辦單位每一筆測試資料的輸入了，請你幫忙計算一下，對於某一筆輸入，小 Y 有多少機率會答對，或者答對的機率太低了。

Input

輸入第一行包含兩個非負整數 N, M ，以空白間隔，分別代表 P 的長度與線索的數量。

接下來 M 行，第 i 行會是 $a_i > b_i$ 或是 $a_i < b_i$ ，分別表示這個線索是 $p_{a_i} > p_{b_i}$ 或是 $p_{a_i} < p_{b_i}$ 。

- $1 \leq N \leq 10^5$
- $0 \leq M \leq 10^5$
- $1 \leq a_i \leq N$
- $1 \leq b_i \leq N$

Output

如果完全沒有任何符合所有線索的排列，請你輸出 -1 ，代表根本沒有存在符合所有線索的排列 P 。如果答對的機率小於 1%，請你輸出 0，代表你建議小 Y 不要用隨機來唬爛了。否則如果答對的機率是 $x\%$ ，請你輸出 x 。

你的答案的絕對或相對誤差不超過 10^{-6} 都會被當作正確。

Sample Input 1	Sample Output 1
3 1 1 < 2	33.3333333333
Sample Input 2	Sample Output 2
7 6 2 > 1 2 < 3 3 < 4 1 > 5 4 < 6 7 > 1	20.0000000000
Sample Input 3	Sample Output 3
3 3 1 < 2 2 < 3 3 < 1	-1.0
Sample Input 4	Sample Output 4
5 0	0.0

G. 陣列刪除

Problem ID: deleting

這天，你在趕著去參加 NPSC (National Party of Super Cat) 的路上，就在你快抵達會場時，一隻巨大的貓貓神突然降臨在你的面前。

「用最小的花費，刪除掉整個陣列吧！」

這麼說著的貓貓神拿出了一個 N 個元素的陣列並擺在你眼前，祂向你說明了規則：

- 你每次可以選擇一對相鄰的元素對將其刪除，並花費「兩個元素的數值最小值」。
- 刪除後，陣列將會合併起來。
- 你必須執行 $\frac{N}{2}$ 個操作來將整個陣列刪除，並且所有操作的花費和要最小。

不快一點完成貓貓神的任務的話，你就見不到可愛的貓咪們了，而經由你苦苦哀求後，貓貓神決定讓你找出最小的花費和即可，快點完成祂的任務並投向貓咪的懷抱吧！

Input

輸入的第一行有一個正整數 N ，代表陣列的大小。

第二行有 N 個以空格分開的正整數 a_1, a_2, \dots, a_N ，代表陣列的內容， a_i 代表第 i 個元素的數值。

- $1 \leq N \leq 2 \times 10^5$
- N 是偶數
- $1 \leq a_i \leq 10^9$

Output

輸出一行一個正整數，代表欲將整個陣列刪除的最小花費。

Sample Input 1	Sample Output 1
4 1 3 2 4	3

Sample Input 2	Sample Output 2
8 1 9 7 6 8 7 6 3	16

Sample Input 3	Sample Output 3
6 3 1 4 1 5 9	5

H. 殿王愛字串

Problem ID: string

殿王是個天才兒童，他在一個月大的時候就學會數數、六個月大的時候就學會乘法跟除法、一歲時學會寫程式、一歲又六個月時養了可愛的拉不拉多、一歲又十個月時養了可愛的貓咪、兩歲時發明了「吃餅乾」的遊戲，三歲又三個月大時成功的對貓咪做了排序，三歲又四個月時創立的飲料王國，販賣了「QQ捏捏好喝到殿王茶」。現在要講的是殿王三歲六個月大時的故事。

在殿王三歲又六個月大時，他在地板上檢到一組字串 A, B 。

因為 A, B 兩個字串長的不一樣，所以愛好平等的殿王就很生氣，決定要把 B 經由一系列的操作變成 A 。

他每次會選定 B 字串上的一個字元，並且把他刪掉。也就是說，他每次會選定 i 使得 $0 \leq i < |B|$ ，令 $B' = B_0B_1B_2 \cdots B_{i-1}B_{i+1}B_{i+2} \cdots B_{|B|-1}$ ，然後把 B 變成 B' 。

已知殿王永遠找得到一個方法把 B 變成 A ，現在殿王想知道，在把 B 變成 A 的所有方法中，若將所有過程中出現的字串蒐集起來，究竟有多少種長的不一樣的字串呢？

字串 X, Y 長的不一樣，若且唯若 $|X| \neq |Y|$ 或存在 $0 \leq i < |X|, X_i \neq Y_i$ 。

因為可能有太多種了，所以殿王只想知道答案除以 998244353 之後的餘數。

Input

輸入的第一行有一個由小寫字母組成的字串 A 。輸入的第二行有一個由小寫字母組成的字串 B 。

- $1 \leq |A| \leq 100$
- $|A| \leq |B| \leq 10000$
- 保證殿王可以將 B 經由一系列操作變成 A

Output

輸出只有一行僅包含一個非負整數，代表答案除以 998244353 之後的餘數。

Notes

在範例測試資料 2 中，殿王將 aba 變成 a 的過程中可以是下列三種的其中一種：

- aba → aa → a
- aba → ab → a
- aba → ba → a

不同的字串有 a, aa, ab, ba, aba, 共 5 種。

Sample Input 1	Sample Output 1
a ab	2
Sample Input 2	Sample Output 2
a aba	5

I. 幸福感

Problem ID: happiness

殿王是個天才兒童，他在一個月大的時候就學會數數、六個月大的時候就學會乘法跟除法、一歲時學會寫程式、一歲又六個月時養了可愛的拉不拉多、一歲又十個月時養了可愛的貓咪、兩歲時發明了「吃餅乾」的遊戲，三歲又三個月大時成功的對貓咪做了排序，三歲又四個月時創立的飲料王國，販賣了「QQ捏捏好喝到殿王茶」，三歲又六個月時學會與字串平等相處。現在要講的是殿王三歲又九個月大時的故事。

在殿王三歲又九個月大時，住進了 CSPN 國，在 CSPN 國，有著一種特別的生物：PETS。

殿王認為，與 PETS 一起生活，是一件非常具有幸福感的事情。而在生活的過程中，殿王還發現，如果跟不同的 PETS 一起生活，生活得到的幸福感會特別高！

殿王想要量化自己的幸福感，因此他對 CSPN 國的 PETS 做出了一些分類。在 CSPN 國裡面，總共有 N 個 PETS，編號從 1 到 N ，對於第 i 個 PETS，殿王會給他一個介於 $[1, N]$ 之間的整數 a_i ，代表這個 PETS 的種類。

接著，殿王要來定義跟一群 PETS 生活可以得到的幸福感。假設殿王跟第 L 個 PETS 到第 R 個 PETS 一起生活，他得到的幸福感的計算方式如下：

- 把 a_L, a_{L+1}, \dots, a_R 這些數字蒐集起來
- 如果某種數字出現超過一次，把那種數字刪除到只剩一個
- 剩下數字的數量，就是殿王得到的幸福感

例如，如果殿王跟種類為 $[1, 2, 1]$ 的 PETS 一起生活的話，得到的幸福感是 2，跟種類為 $[1, 2]$ 的 PETS 一起生活的話，得到的幸福感也是 2，但跟種類為 $[1]$ 的 PETS 一起生活的話，得到的幸福感便只有 1。

在接下來的 Q 天內，殿王每天都想著要跟哪些 PETS 一起生活。在第 i 天中，殿王會特別想跟第 L_i 個 PETS 到第 R_i 個 PETS 中的其中一些 PETS 生活。但是，有選擇障礙的殿王，常常不知道要怎麼選擇。於是，他索性隨機選擇滿足 $L_i \leq l \leq r \leq R_i$ 的 (l, r) ，並且跟第 $l, l+1, \dots, r$ 個 PETS 生活。每個滿足條件的 (l, r) 都有相同的機率被選擇。

而殿王也知道，每天這樣子隨機的選擇，很有可能會帶給他隨機的幸福感。於是他很好奇，每天這樣子生活下來，每天個別得到的幸福感期望值是多少？

但是殿王特別懶得計算這種複雜的問題，於是他找上了你，打算請你告訴他每天的幸福感期望值。而他也知道計算期望值時，浮點數誤差的問題很麻煩，於是對於第 i 天，你只要告訴他第 i 天的期望值乘上 $\frac{(R_i - L_i + 1)(R_i - L_i)}{2}$ 就可以了。可以證明，如此一來，你需要告訴殿王的數字全部都是整數。

Input

輸入的第一行包含兩個正整數 N, Q ，分別代表 PETS 的數量，以及殿王接下來要度過的天數。

接下來的一行，包含 N 個正整數 a_1, a_2, \dots, a_N ， a_i 代表第 i 個 PETS 的種類。

接下來的 Q 行，第 i 行包含著兩個正整數 L_i, R_i ，代表殿王在第 i 天特別想跟第 L_i 個 PETS 到第 R_i 個 PETS 中的其中一些 PETS 生活。

- $1 \leq N \leq Q \leq 3 \times 10^5$
- $1 \leq a_i \leq N$
- $1 \leq L_i \leq R_i \leq N$

Output

請輸出 Q 行，第 i 行請輸出第 i 天殿王希望你告訴他的數字。

Sample Input 1	Sample Output 1
3 4	9
1 2 1	4
1 3	1
2 3	9
1 1	
1 3	

Sample Input 2	Sample Output 2
7 5 2 4 7 4 3 3 1 1 7 3 6 1 5 2 6 3 4	66 17 31 28 4

This page is intentionally left blank.