

A. 青椒大戰 1 - 特殊選才的惡夢

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

前情提要:

(以下內容不閱讀也不影響解題，建議大家題目都想不出來心煩氣燥時再來看故事。)

「特殊選才」是台灣的學生升大學的一個管道，從某些學校公開的錄取名單可以發現，藉由特殊選才錄取資工系的學生幾乎都是競程選手。

其中，幾乎所有競程選手都報名了「青麻大學」和「椒麻大學」的資工系，dreamoon 老師也有許多參加特殊選才的學生報名了這兩所大學。因此，dreamoon 老師每天都因擔心學生上不了大學而睡不好覺，就連放榜後，睡夢中都在擔心備取的學生們能不能被錄取...

某天，dreamoon 老師夢到了最終「青麻大學」和「椒麻大學」的最終錄取名單裡沒半個自己的學生，於是就大喊「-1」並驚醒了！

醒後 dreamoon 還記得很清楚夢中有哪些人是最終被錄取的，為了證明自己做的夢不可能是真的，dreamoon 想要寫個程式，驗證這個名單的結果有誤，但 dreamoon 已經被嚇的神智不清，請大家幫忙寫程式，驗證是否存在一個合理的放棄錄取資格的順序，使得最終被錄取的人恰好是 dreamoon 夢到的名單。細節在真正的題目敘述裡說明。

真正的題目敘述:

兩條分隔線夾著的內容和 B 題「青椒大戰 2 - 惡夢再臨」裡的一樣。

----- 我是分隔線 -----

共有 N 位學生同時報名了「青麻大學」和「椒麻大學」的資訊工程學系特殊選才，這 N 位學生被編號為 $1 \sim N$ ，除了這 N 位學生以外沒有其他學生報名。「青麻大學」和「椒麻大學」根據大家的備審資料以及面試過程給所有學生排名，「青麻大學」排名第 i 的學生編號為 p_i ，在「椒麻大學」排名第 i 的學生編號為 q_i 。「青麻大學」預計錄取 x 位學生，「椒麻大學」預計錄取 y 位學生，保證 $x + y \leq N$ 。

假設「最終的」錄取名單的產生過程如下：

- 首先，兩所學校會各維護一個錄取名單。初始時，「青麻大學」排名前 x 的學生，和「椒麻大學」排名前 y 的學生都在各自的錄取名單上。
- 若在某個時刻，兩所學校維護的錄取名單中，沒有重複的人，那麼此時的錄取名單就是「最終的」錄取名單。（我們假設大家都很安分，所有的人目標都是錄取這兩所大學的其中一所，並且會放棄錄取資格的唯一情況是同時被兩所學校錄取。）
- 只要當前不是上一項描述的情況，代表至少有一位學生同時被兩所大學錄取，此時，將會依序發生以下兩個事件，直到變成上一項描述的情況為止：
 1. 有某個同時在兩所學校的錄取名單裡的學生決定放棄其中一間學校的錄取資格，於是該學生就被從該學校的錄取名單裡移除。
 2. 在 1. 中被放棄資格的學校，就會遞補下一位學生，也就是說，該學校排名最小且不曾被加入錄取名單的學生會被加入該學校的錄取名單，並收到該學校的錄取通知。有辦法證明在 $x + y \leq N$ 的情況下，一定有學生可以被遞補。

請注意，我們假設一次只會有一位學生放棄錄取資格，且遞補的學生收到錄取通知後，才能有下一位學生放棄錄取資格。

----- 我是分隔線 -----

現在給你兩所學校「最終的」錄取名單，請找到一個學生們放棄錄取資格的順序，使得「最終的」錄取名單是給定的名單。

假設過程中共有 m 位學生放棄錄取資格，放棄錄取資格的順序可用一個長度為 m 的正整數序列表示，數列裡第 i 個數代表第 i 個放棄錄取資格的學生編號。

若有多種可能的放棄資格的順序，請輸出字典序最小的解(字典序最小的定義請參照 Note)。若給你的「最終的」錄取名單是不可能存在的，請輸出 -1。

Input

```
N
p1 p2 ... pN
x
Sp
q1 q2 ... qN
y
Sq
```

- N 為學生人數。
- p_i 為「青麻大學」排名第 i 位的學生的編號。
- x 為「青麻大學」預計錄取的學生人數。
- S_p 為長度 N 且由 $N - x$ 個數字字元 '0' 和 x 個數字字元 '1' 組成的字串，若第 i 個字元是 '0'，代表編號 p_i 的學生不在「青麻大學」 「最終的」錄取名單裡，若第 i 個字元是 '1'，代表編號 p_i 的學生在「青麻大學」 「最終的」錄取名單裡。
- q_i 為「椒麻大學」排名第 i 位的學生的編號。
- y 為「椒麻大學」預計錄取的學生人數。
- S_q 為長度 N 且由 $N - y$ 個數字字元 '0' 和 y 個數字字元 '1' 組成的字串，若第 i 個字元是 '0'，代表編號 q_i 的學生不在「椒麻大學」 「最終的」錄取名單裡，若第 i 個字元是 '1'，代表編號 q_i 的學生在「椒麻大學」 「最終的」錄取名單裡。

Output

情形一:

-1

或

情形二:

 m
 $d_1 d_2 \dots d_m$

- 情形一：若輸入給定的兩所學校的「最終的」錄取名單若是不可能存在的，共輸出一行，包含數字 -1 。
- 情形二：若存在至少一個學生們放棄錄取資格的順序，共輸出兩行。 m 是過程中放棄錄取資格的學生數量， d_i 是第 i 位放棄錄取資格的學生的編號。

Constraints

- 兩所學校「最終的」錄取名單裡不存在編號相同的學生。
- $1 \leq x, y < N$ 。
- $x + y \leq N$ 。
- $N \leq 2 \times 10^5$ 。
- 保證 p 和 q 是一個 $1 \sim N$ 的排列。
- S_p 裡恰有 x 個 '1'， $N - x$ 個 '0'。
- S_q 裡恰有 y 個 '1'， $N - y$ 個 '0'。

Scoring

本題共有三組子任務，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆須答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	14	$N \leq 1000$, 保證答案不是 -1
2	16	$N \leq 1000$
3	70	無額外限制

Examples

input
2
1 2
1
01
2 1
1
01
output
-1

input
2
1 2
1
10
2 1
1
10
output
0

input
5
5 4 3 2 1
2
01100
2 4 5 1 3
2
10100
output
2
4 5

input
4
3 2 1 4
2
1001

```
3 2 1 4
2
0110
```

output

```
3
2 3 1
```

Note

定義數列 $a = (a_1, a_2, \dots, a_n)$, 數列 $b = (b_1, b_2, \dots, b_m)$ 。

若數列 a 的字典序比數列 b 的字典序還小, 若且唯若以下兩件事情發生其中一件:

1. 數列 a 是數列 b 的前綴。
2. 令 i 是最小的正整數滿足 $a_i \neq b_i$, $a_i < b_i$ 。

B. 青椒大戰 2 - 惡夢再臨

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

前情提要:

(以下內容不閱讀也不影響解題，建議大家題目都想不出來心煩氣燥時再來看故事。)

有一就有二，有二就有三，自從第一次做了特殊選才的惡夢後，dreamoon 老師天天都在做一樣的惡夢，惡夢裡唯一不同的地方是每次最終錄取名單都長得不一樣。雖然夢醒之後 dreamoon 老師藉由程式驗證了這些天夢到的最終錄取名單都是不可能發生的，但 dreamoon 老師希望能找到簡單的方法，一眼辨別出此錄取名單是否有機會發生，這一次，dreamoon 老師終於憑自己的力量找到簡單的方法辨別了！並且還能計算出總共到底有多少種可能的最終錄取名單！現在，dreamoon 老師決定把這個問題出在 TOI 模擬賽考考大家，讓大家一起體會可怕的惡夢：)

真正的題目敘述:

兩條分隔線夾著的內容和 A 題「青椒大戰 1 - 特殊選才的惡夢」裡的一樣。

----- 我是分隔線 -----

共有 N 位學生同時報名了「青麻大學」和「椒麻大學」的資訊工程學系特殊選才，這 N 位學生被編號為 $1 \sim N$ ，除了這 N 位學生以外沒有其他學生報名。「青麻大學」和「椒麻大學」根據大家的備審資料以及面試過程給所有學生排名，「青麻大學」排名第 i 的學生編號為 p_i ，在「椒麻大學」排名第 i 的學生編號為 q_i 。「青麻大學」預計錄取 x 位學生，「椒麻大學」預計錄取 y 位學生，保證 $x + y \leq N$ 。

假設「最終的」錄取名單的產生過程如下：

- 首先，兩所學校會各維護一個錄取名單。初始時，「青麻大學」排名前 x 的學生，和「椒麻大學」排名前 y 的學生都在各自的錄取名單上。
- 若在某個時刻，兩所學校維護的錄取名單中，沒有重複的人，那麼此時的錄取名單就是「最終的」錄取名單。(我們假設大家都很安分，所有的人目標都是錄取這兩所大學的其中一所，並且會放棄錄取資格的唯一情況是同時被兩所學校錄取。)
- 只要當前不是上一項描述的情況，代表至少有一位學生同時被兩所大學錄取，此時，將會依序發生以下兩個事件，直到變成上一項描述的情況為止：
 1. 有某個同時在兩所學校的錄取名單裡的學生決定放棄其中一間學校的錄取資格，於是該學生就被從該學校的錄取名單裡移除。
 2. 在 1. 中被放棄資格的學校，就會遞補下一位學生，也就是說，該學校排名最小且不曾被加入錄取名單的學生會被加入該學校的錄取名單，並收到該學校的錄取通知。有辦法證明在 $x + y \leq N$ 的情況下，一定有學生可以被遞補。

請注意，我們假設一次只會有一位學生放棄錄取資格，且遞補的學生收到錄取通知後，才能有下一位學生放棄錄取資格。

----- 我是分隔線 -----

請問「最終的」錄取名單有幾種不同的可能？由於答案很大，請輸出答案除以 998,244,353 的餘數。

Input

N
$p_1 p_2 \dots p_N$
x
$q_1 q_2 \dots q_N$
y

- N 為學生人數。
- p_i 為「青麻大學」排名第 i 位的學生的編號。保證 p 是一個 $1 \sim N$ 的排列。
- x 為「青麻大學」預計錄取的學生人數。
- q_i 為「椒麻大學」排名第 i 位的學生的編號。保證 q 是一個 $1 \sim N$ 的排列。
- y 為「椒麻大學」預計錄取的學生人數。

Output

ans

- $answer$ 為一個整數，代表答案除以 998,244,353 的餘數。

Constraints

- $N \leq 2 \times 10^5$ 。
- $1 \leq x, y < N$ 。
- $x + y \leq N$ 。

Scoring

本題共有四組子任務，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆須答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	22	$N \leq 20$
2	21	$N \leq 100$
3	19	$N \leq 5000$

Examples

input
2 1 2 1 1 2 1
output
2

input
2 1 2 1 2 1 1
output
1

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

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

C. 圓桌舞事

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

前情提要：

(以下內容不閱讀也不影響解題，建議大家題目都想不出來心煩氣燥時再來看故事。)

與 dreamoon 老師不同，小 AA 大老闆每天都期待著美好的事情發生。某一天下午，在 dreamoon 老師緊張地給學生培訓 TOI 的時候，她卻在沙發上睡得很香，而且還做起了白日夢。在夢裡，學生們在比賽的時候都發揮出他們自身的實力，取得了不錯的成績，小 AA 大老闆非常開心，於是邀請他們晚餐一起去吃海底撈。由於時間倉促，小 AA 大老闆並沒有一個個學生單獨邀請，而是在群組裡 @everyone，並告訴大家時間和地點。晚上到了吃飯的時候，小 AA 大老闆才意識到：平常都在線上上課，學生們都不熟，而且大多數人還是社恐。他們都只和自己學校的人坐在一起，有一些人甚至沒有出現...

夢醒之後，小 AA 大老闆決定，暑假開辦線下課，創造更多老師和學生、學生和學生互動的機會！

真正的題目敘述：

有 n 位學生圍著圓桌吃飯，座位按照順時針編號為 $1 \sim n$ ，座位 i 和座位 $i+1$ 相鄰 ($1 \leq i < n$)，且座位 1 和座位 n 相鄰。

來自同一個學校的人都坐在連續的位置。這意思是說，若學生 A 和學生 B 是來自同一個學校，那麼從 A 開始順時針到 B 間所有人也是來自同一個學校，或是從 A 開始逆時針到 B 間所有人也是來自同一個學校。

現在每個人都要告知大家 n 個人中，和他來自同一間學校的學生有多少人。現在有些人已經回答了、有些還沒。給你 n 個整數 a_1, a_2, \dots, a_n ，若 $a_i = 0$ ，代表坐在座位 i 的學生還沒回答，若 $a_i > 0$ ，代表坐在座位 i 的人回答是 a_i 。請問，在只知道目前的資訊的情況下，最終每個人的回答共有多少種可能？有於答案可能很大，請輸出答案除以 998,244,353 的餘數。若當前的資訊已存在矛盾，請輸出 0。

最後再次強調，我們問的是最終每個人的回答有多少種，而不是哪些人來自同所學校的組合有多少種，可參考範例測試資料 3 的說明來更瞭解題意。

Input

```
n
a1 a2 ... an
```

- n 為學生的人數。
- $a_i = 0$ 代表坐在座位 i 的學生還沒回答，若 $a_i > 0$ 代表坐在座位 i 的學生的回答是 a_i 。

Output

```
ans
```

- ans 是一個整數，代表答案除以 998,244,353 的餘數，若當前的資訊已存在矛盾， $ans = 0$ 。

Constraints

- $1 \leq n \leq 5000$ 。
- $0 \leq a_i \leq n$ 。

Scoring

本題共有三組子任務，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆須答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	25	$n \leq 200, a_n = 1$
2	25	$n \leq 200$
3	50	無額外限制

Examples

input
5
0 0 2 0 1
output
3

input
2
2 1
output
0

input
4
0 2 2 0
output
2

Note

關於範例測試資料 3 的解釋如下：

最終所有人可能的回答只有兩種情形，分別是 1 2 2 1 和 2 2 2 2。

D. 天下沒有不散的宴席

time limit per test: 1.5 seconds
 memory limit per test: 256 megabytes
 input: standard input
 output: standard output

前情提要:

(作者外出取材中...休刊一題)

真正的題目敘述:

有一個餐會共有 N 個學生參加，學生被編號為 $1 \sim N$ ，每個學生的個性可用一個小於 2^{20} 的非負整數表示，編號 i 的學生個性為 a_i ，保證每個學生的個性都不相同。編號 i 的學生和編號 j 的學生是朋友若且唯若 a_i 和 a_j 寫成二進制至多只有 2 個位數不一樣。例如說個性為 3 (二進制是 11_2) 和個性為 5 (二進制是 101_2) 的學生是朋友，但是個性 3 和個性 4 (二進制是 100_2) 的學生不是朋友。

餐會中的學生們會組成很多小團體，每位學生恰屬於一個小團體。編號 x 的學生和編號 y 的學生在同一個小團體 若且唯若 餐會中存在 k 個人 ($k \geq 2$) 編號依序為 u_1, u_2, \dots, u_k ，滿足 $u_1 = x, u_k = y$ 且對於所有 $1 \leq i < k$ ，編號 u_i 和編號 u_{i+1} 的學生是朋友。(換句話說，若把每位學生想像成圖論上的點，兩個學生是朋友就想像成兩個點之間存在邊，那麼小團體就是對應到圖論中連通元件的概念。)

定義一個小團體的和諧度為此小團體裡每個學生的朋友數量乘積，例如說，若某個小團體裡有 4 個學生，個性分別是 1, 2, 3, 13，那麼個性 1, 2, 3 的 3 個學生互為朋友，但個性 13 的學生只和個性 1 的學生是朋友，所以這 4 人朋友的數量分別是 3, 2, 2, 1，於是此小團體的和諧度是： $3 \times 2 \times 2 \times 1 = 12$ 。若一個學生沒有任何朋友，他自己也是個小團體，不過呢，根據以上定義，只有一個學生的小團體和諧度是 0。

餐會結束時，學生們會一個一個的離開餐廳，告訴你 N 個學生的離開順序，第 i 個離開的是編號 d_i 的學生，請對於所有 $1 \leq i \leq N$ ，都請計算一個數 S_i 代表在編號 d_i 的學生離開之前，在僅考慮所有還在餐廳裡的學生，他們所形成的所有小團體的和諧度的總和。

請注意，一個一個小團體可能因為其中一個人離開，就份列為多個小團體，請參照 Note 裡範例測試資料一的解釋。

由於答案可能很大，請輸出 S_i 除以 998,244,353 的餘數即可。

Input

```
N
a1 a2 ... aN
d1 d2 ... dN
```

- N 為學生人數。
- a_i 為編號 i 的學生的個性。
- d_i 為第 i 個離開餐會的學生的編號。

Output

```
ans1
ans2
...
ansN
```

- ans_i 為一個整數，代表 S_i 除以 998,244,353 的餘數。

Constraints

- $N \leq 8 \times 10^4$ 。
- $0 \leq a_i < 2^{20} = 1,048,576$ 。
- 若 $i \neq j$ ，那麼 $a_i \neq a_j$ 。
- $1 \leq d_i \leq N$ ， d 是一個 $1 \sim N$ 的排列。

Scoring

本題共有三組子任務，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆須答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	14	$N \leq 200$
2	27	$N \leq 5000$
3	59	無額外限制

Examples

input	output
4 1 2 3 13 1 2 3 4	

```
12
1
0
0
```

input

```
5
3 2 1 7 6
2 4 1 5 3
```

output

```
576
36
2
0
0
```

Note

在範例測試資料一中，最初餐會有 4 個學生，個性分別是 1, 2, 3, 13，其中編號 1, 2, 3 的 3 個學生互為朋友，但編號 4 的學生只和編號 1 的學生是朋友，所以在編號 1 的學生離開之前，和諧度是 $3 \times 2 \times 2 \times 1 = 12$ 。

接著 1 號學生離開了，原來的小團體被拆成兩個小團體，其中一個是由編號 2 和編號 3 的學生組成，另一個是編號 4 的學生自己是一個小團體，所以兩個小團體的和諧度總和是 $1 \times 1 + 0 = 1$ 。

在這之後，都不存在兩個人以上的小團體了，所以編號 3 和編號 4 的學生離開前和諧度總合都是 0。

E. 大聲說討厭的勇氣

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

前情提要：

<https://leetcode.com/> 是在軟體工程師間非常著名的線上解題系統，很多人在準備應徵軟體工程師相關的工作時，都會在此平台上做題目練習。Leetcode 和 Codeforces、Atcoder 等競程圈比較著名的平台有個不一樣的地方：解題者並不是寫一個主函式從標準輸入讀入測試資料，而是實作一會接受測試資料的函式，讓這個函式回傳答案。因此，做題者想要測試自己的程式碼的方法就比較麻煩，尤其當傳入函式的物件是一棵二元樹時，在平台上測試的人必須把測試資料用 Leetcode 指定的方式，壓縮成一個由一些整數和 null 組成的陣列。今天 dreamoon 老師為了讓大家熟悉這種壓縮二元樹的方式，所以出個相關的題考考大家。

真正的題目敘述：

<https://leetcode.com/> 這個線上解題系統中，當測試資料是一棵二元樹(binary tree) 時，會把測試資料用 [1,null,2,3] 這樣的格式表示，英文的解釋請見此[連結](#)，中文的解釋則請見此[連結](#)，請自行閱讀理解。

輸入會給你兩個正整數 x 和 y ，請構造一棵二元樹，此二元樹使用 Leetcode 的表示方法時必須滿足以下條件：

1. 中括號裡有 y 個元素，每個元素都是 1 或 null
2. 中括號裡第 x 個元素和第 y 個元素都不是 null
3. 中括號裡第 x 個元素對應到的節點的左子節點是第 y 個元素對應到的節點。

若有多組解，可輸出任意一個，但若不存在這樣的二元樹，請輸出 ":(" (雙引號不用輸出)。

Input

$x y$

- x, y 皆為正整數，代表你構造的二元樹必須滿足中括號裡第 x 個元素對應的節點的左子節點是第 y 個元素。

Output

S

- S 為一字串，若有解， S 是你構造出的滿足題目條件的二元樹使用 Leetcode 的格式(不能輸出任何空白字元)儲存在字串裡，有多解時，可以輸出任一個。若沒有任何解， S 是 ":(" (雙引號不用輸出)。

輸出一行，若有解，請使用 Leetcode 的格式(不能輸出任何空白字元)輸出你構造出的滿足題目條件的二元樹，有多解時，可以輸出任一個。若沒有任何解，請輸出 ":(" (雙引號不用輸出)。

若輸出字串長度超過 10^6 會直接被判作 Wrong Answer。

Constraints

- $1 \leq x < y \leq 10^5$ 。

Scoring

本題共有三組子任務，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆須答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	13	$y \leq 8$
2	43	$y \leq 200$
3	44	無額外限制

Examples

input
1 3
output
:(

input
6 10
output
[1,1,1,null,1,1,1,null,1,1]

Note

提示：如下圖，可在 [Leetcode 94. Binary Tree Inorder Traversal](#) 這一題把測試資料改成你程式產生的輸出，就可以看到圖像化後的二元樹。

Testcase Result

Case 1 Case 2 Case 3 + 修改這裡

root =

[1,1,1,null,1,1,1,null,1,1]

