**D-Bán bánh đồng xu tự động**

Sau khi mở start-up bán bánh đồng xu thì Thắng và Tâm đã thu được một ít lãi và muốn nâng cấp cửa hàng của mình lên bằng cách lắp đặt một hệ thống tự động làm bánh và đưa đến cho khách hàng. Mẫu mã của bánh cũng đa dạng hơn nên được đánh số để phân biệt các mẫu bánh (mã là các số nguyên dương). Vì muốn bánh đưa đến cho khách hàng vẫn còn nóng hổi vừa thổi vừa ăn (một phần cũng là do nghèo nên không mua được máy xịn) nên mỗi máy làm bánh sẽ được chuyển thẳng tới chỗ bán và đưa bánh cho khách hang.

Cửa hàng có k mẫu bánh. Mỗi mẫu bánh sẽ có 1 máy làm riêng. Có n khách hàng muốn mua bánh trong 1 ngày, mỗi khách hàng sẽ mua m cái bánh. Ban đầu, các máy được xếp theo thứ tự ngẫu nhiên (do hôm trước bán xong nên có thứ tự như vậy), mỗi khi có một yêu cầu làm bánh, hệ thống sẽ vừa làm vừa chuyển chiếc máy đó lên đầu tiên (vị trí bán cho khách hàng). Mỗi lần chuyển qua 1 máy thì sẽ mất 1 giây, cộng thêm 1 giây để đưa bánh ra cho khách nữa.

Là một con người say đắm trong tình yêu, người yêu bạn nhờ bạn mua một vài chiếc bánh. Nhưng xui cho bạn là bạn lại là người đến sau và phải xếp hàng ở cuối cùng. Bạn hãy tính thời gian để hệ thống kia có thể phục vụ hoàn tất cho bạn để canh thời gian kể với người yêu (không thì bạn ấy sẽ dỗi).

**Input:**

Dòng đầu tiên chứa ba số nguyên n , m và k ( 1 ≤  n ,  k  ≤ 100, 1 ≤  m  ≤  k ) — số lượng khách hàng, số lượng bánh mỗi khách hàng muốn mua và tổng số mặt hàng tại hệ thống.

Dòng tiếp theo chứa k số nguyên pi ( 1 ≤  pi  ≤  k ) biểu thị mẫu của các mặt hàng. Các mã được đánh số nguyên từ 1 đến k .

Mỗi dòng trong số n dòng tiếp theo chứa m số nguyên riêng biệt thể hiện loại bánh mà khách hàng đó muốn mua.

**Output:**

In ra thời gian của từng vị khách hàng, mỗi số cách nhau 1 dấu cách.

Nếu không có kết quả gì, xuất ra -1.

Examples:

**input**

2 2 5  
3 4 1 2 5  
1 5  
3 1

**output**

8 6

Giải thích:

Khách hàng 1 muốn món bánh 1 và 5 .

Hệ thống mất 2 giây để đưa máy 1 lên đầu và 1 giây để đưa bánh 1 cho khách, vì vậy vị trí mới của máy là: [1, 3, 4, 2, 5] .

Hệ thống mất 4 giây để đưa máy 5 lên đầu và 1 giây để đưa bánh 1 cho khách, vì vậy vị trí mới của máy là: [5, 1, 3, 4, 2] .

Thời gian dành cho khách hàng đầu tiên là 3 + 5 = 8 .

Khách hàng 2 muốn các mặt hàng 3 và 1 .

Hệ thống mất 2 giây để đưa máy 3 lên đầu và 1 giây để đưa bánh 1 cho khách, vì vậy vị trí mới của máy là: [3, 5, 1, 4, 2] .

Hệ thống mất 2 giây để đưa máy 1 lên đầu và 1 giây để đưa bánh 1 cho khách, vì vậy vị trí mới của máy là: [1, 3, 5, 4, 2] .

Thời gian dành cho khách hàng thứ hai là 3 + 3 = 6 .

**Test case #1:**

**input**

3 4 5

4 2 5 1 3

5 4 2 4

4 2 2 5

5 1 4 4

**output**

10 7 10

**Test case #2:**

**input**

10 10 10

4 9 5 1 3 10 2 7 8 6

3 6 6 8 3 7 7 3 7 1

8 10 5 9 2 10 9 5 10 1

1 9 9 5 2 5 2 8 5 4

1 2 2 2 1 2 3 5 5 6

5 4 4 4 6 1 7 1 4 3

9 10 4 2 7 4 10 4 3 2

2 4 1 1 4 1 8 3 6 8

7 9 9 6 8 2 1 9 3 4

9 4 8 8 9 1 6 1 8 9

7 3 4 5 7 9 7 9 6 1

**output**

52 61 38 42 41 59 43 61 37 61

**Test case #3:**

**input**

0 10 10

3 6 9 2 8 1 7 10 4 5

**output**

-1

**Test case #4:**

**input**

10 0 5

2 4 1 5 3

**output**

0 0 0 0 0 0 0 0 0 0

**Test case #5:**

**input**

100 15 20

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

17 6 13 3 2 6 14 9 5 16 20 12 3 5 19

18 6 14 4 4 7 2 2 11 3 1 20 5 3 18

10 7 1 10 13 10 3 6 16 10 11 18 16 1 15

5 1 4 13 16 5 5 5 8 9 7 9 9 19 1

9 13 4 3 8 5 17 13 3 11 12 1 5 11 15

9 3 13 11 19 7 11 7 11 4 1 10 15 13 6

2 8 18 2 18 17 3 12 8 12 2 2 10 3 14

16 17 5 6 13 3 3 17 15 16 20 14 6 17 8

…

**output**

179 172 148 135 152 137 156 158 183 174 113 154 111 171 181 171 202 …

**Test case #6:**

**input**

11 4 4

4 2 3 1

3 4 4 1

2 4 4 4

4 3 3 2

3 4 4 4

4 3 3 3

4 4 1 4

1 3 1 2

1 1 1 4

4 1 2 4

2 3 2 2

2 2 3 4

**output**

10 9 9 7 5 9 11 8 9 9 7

**Test case #7:**

**input**

15 0 0

**output**

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

**Test case #8:**

**input**

4 17 20

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

15 11 8 2 14 1 2 16 13 13 2 17 1 20 12 4 6

5 1 17 8 3 15 19 6 6 20 5 12 5 11 14 18 1

8 1 7 3 1 6 18 18 4 2 19 3 3 10 5 10 7

11 7 4 11 13 6 9 17 12 19 6 10 19 15 17 8 15

**output**

190 197 164 186

**Test case #9:**

**input**

0 0 0

**output**

-1

**Test case #10:**

**input**

5 6 7

6 3 1 5 2 4 7

1 7 3 1 2 7

1 1 6 2 6 5

2 2 4 6 4 2

2 5 4 7 5 7

2 5 6 6 4 5

**output**

27 21 20 19 21