- **EASY**

- **Max Score: 30 Points**

Compare Two Linked lists

Compare the data in the nodes of the linked lists to check if they are equal. If all data attributes are equal and the lists are of same length, print `1`. Otherwise, print `0`.

Input Format

The first line contains two integers `m` and `n`, the number of nodes in the first linked list and second linked list respectively.

Second line contains `m` spaced integers, each a value for a data attribute of list 1.

Third line contains `n` spaced integers, each a value for a data attribute of list 2.

Output Format

Compare the two linked lists and print `1` if the lists are equal. Otherwise, print `0`.

Example 1

Input

```
2 1
1 2
1
```

Output

```
0
```

Explanation

Linked list 1= 1->2->NULL

Linked list 2= 1->NULL

It is clearly visible that the two linked lists are not equal. Therefore, print 0.

Example 2

Input

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

Output

```
1
```

Explanation

Linked list 1= 1->2->3->4->NULL

Linked list 2= 1->2->3->4->NULL

It is clearly visible that the two linked lists are equal. Therefore, print 1.

Constraints

`1<=n,m<=1000`

`1<=llist1[i],llist2[i]<=1000`

My code

```java
// in java
import java.util.*;
import java.lang.*;
import java.io.*;


class Node
{
    int data;
    Node next;

    Node(int data, Node next)
    {
        this.data = data;
        this.next = next;
    }

    Node() {}
}


public class Main
{
  static int com(Node a,Node b)
  {
    while(a!=null) //both node are same length no need to both.
      {
        if(a.data !=b.data)
        {
         return 0;
```

```java
        }
        a=a.next;
        b=b.next;
      }
    return 1;
  }
      public static void main (String[] args) throws
java.lang.Exception
      {
            //your code here
    Scanner s=new Scanner(System.in);
      Node a=null,b=null;

      int n=s.nextInt();
      int m=s.nextInt();
      int arr[]=new int[n];
for(int i=0;i<n;i++)
  arr[i]=s.nextInt();
 for(int i=n;i>0;i--)
    a=new Node(arr[i-1], a);


      int arrb[]=new int[m];
      for(int i=0;i<m;i++)
     arrb[i]=s.nextInt();

for(int i=m;i>0;i--)
   b=new Node(arrb[i-1], b);

    if(n!=m)
```

```java
        {
            System.out.print("0");
            return;
        }
            System.out.print(com(a,b));
            }
    }
```