

/*

Given a linked list of nodes such that it may contain a loop.

A loop here means that the last node of the link list is connected to the node at position X(1-based index). If the link list does not have any loop, X=0.

Remove the loop from the linked list, if it is present, i.e. unlink the last node which is forming the loop.

N = 3

value[] = {1,3,4}

X = 2

1The link list looks like

1 -> 3 -> 4

^ |
|____|

A loop is present. If you remove it successfully, the answer will be 1.

*/

// Solution

import java.util.*;

// Creating the structure of the node

class Node{

int data;

Node next;

Node(int data){

this.data = data;

next = null;

}

```
}
```

```
public class RemoveLoopLinkedList {
```

```
// Declaring the head of the linked list globally and initialize it as null
```

```
static Node head = null;
```

```
// Creating the insert function for inserting the elements into the linked list
```

```
public static void insert(int data){
```

```
    Node newNode = new Node(data);
```

```
    if(head == null){
```

```
        head = newNode;
```

```
    }
```

```
    else{
```

```
        Node temp = head;
```

```
        while(temp.next != null){
```

```
            temp = temp.next;
```

```
        }
```

```
        temp.next = newNode;
```

```
    }
```

```
}
```

```
// Creating makeLoop() function for making loop in our linked list
```

```
// by linking the last node with the node having position X (1 - based index).
```

```
public static void makeLoop(int pos){
```

```

Node lastNode = head;
while(lastNode.next != null){
    lastNode = lastNode.next;
}
Node temp = head;
while(pos-- > 1){
    temp = temp.next;
}
lastNode.next = temp;
}

```

// Creating removeLoop() function for removing the loop in our
// linked list which was created the makeLoop() function.

```

public static void removeLoop(){
    ArrayList<Node> nodeList = new ArrayList<>();
    Node temp = head;
    nodeList.add(head);
    while(temp.next != null){
        if(nodeList.contains(temp.next)){
            temp.next = null;
            break;
        }
        nodeList.add(temp.next);
        temp = temp.next;
    }
    nodeList.clear();
}

```

```
}
```

```
// Creating detectLoop() function for checking wheather our  
// linked list have any loop or not,  
// if it have a loop means return true, otherwise return false.
```

```
public static boolean detectLoop(){  
    ArrayList<Node> nodeList = new ArrayList<>();  
    Node temp = head;  
    nodeList.add(head);  
    while(temp.next != null){  
        if(nodeList.contains(temp.next)){  
            nodeList.clear();  
            return true;  
        }  
        nodeList.add(temp.next);  
        temp = temp.next;  
    }  
    nodeList.clear();  
    return false;  
}
```

```
public static void main(String[] args) {  
    Scanner scan = new Scanner(System.in);  
    int N = scan.nextInt(); // getting the no of nodes  
    int values[] = new int[N];  
    // getting values for the linked list  
    for (int i = 0; i < N; i++) {
```

```

        values[i] = scan.nextInt();
        insert(values[i]);
    }
    int X = scan.nextInt();

    makeLoop(X); // creating a loop in linked list
    removeLoop(); // removing loop from linked list

    // checking for loop in linked list
    if(detectLoop())
        System.out.println(0); // linked list have a loop
    else
        System.out.println(1); // linked list does not have a loop
    scan.close();
}
}

```

// Output:

```

PROBLEMS 30 OUTPUT TERMINAL DEBUG CONSOLE
PS C:\Users\asakt\Desktop\Internship> cd "c:\Users\asakt\Desktop\Internship\Assignments\Remove Loop in Linked List\"
; if ($?) { javac RemoveLoopLinkedList.java } ; if ($?) { java RemoveLoopLinkedList }
3
1 3 4
2
1
PS C:\Users\asakt\Desktop\Internship\Assignments\Remove Loop in Linked List> cd "c:\Users\asakt\Desktop\Internship\Assignments\Remove Loop in Linked List\" ; if ($?) { javac RemoveLoopLinkedList.java } ; if ($?) { java RemoveLoopLinkedList }
4
1 8 3 4
0
1
PS C:\Users\asakt\Desktop\Internship\Assignments\Remove Loop in Linked List> cd "c:\Users\asakt\Desktop\Internship\Assignments\Remove Loop in Linked List\" ; if ($?) { javac RemoveLoopLinkedList.java } ; if ($?) { java RemoveLoopLinkedList }
4
1 2 3 4
1
1
PS C:\Users\asakt\Desktop\Internship\Assignments\Remove Loop in Linked List>

```