APPROACH

Here, as we need to solve the given problem using O(1) space, therefore while traversing through the array, we need to store two things at each index:

- Its current value (arr[i])
- Its transformed value (arr[arr[i]])

Now, we have to store its transformed value obviously because that's what we are supposed to return. But along with that we also need to store its current value because for index after i whose value equals i, when they refer to this index i, they should be able to get the initial value (arr[i]) and not the updated value(arr[arr[i]]).

Now, we can achieve this using mathematical trick.

We all know:

*Dividend = Divisor * Quotient + Remainder;*

So when we divide dividend by Quotient later, we will obtain Divisor.

So, we will first modify our arr[i] as

*arr[i]   = (arr[arr[i]]%n)*n +   arr[i];*

*Divivend =   Divisor*Quotient + Remainder*

Later when we do arr[i] / n, we get the Divisor that is arr[arr[i]]%n = arr[arr[i]] as all elements are smaller than n.