Approach 3: Floyd's Tortoise and Hare (Cycle Detection)

**Intuition**

The idea is to reduce the problem to [Linked List Cycle II](https://leetcode.com/problems/linked-list-cycle-ii/solution/):

Given a linked list, return the node where the cycle begins.

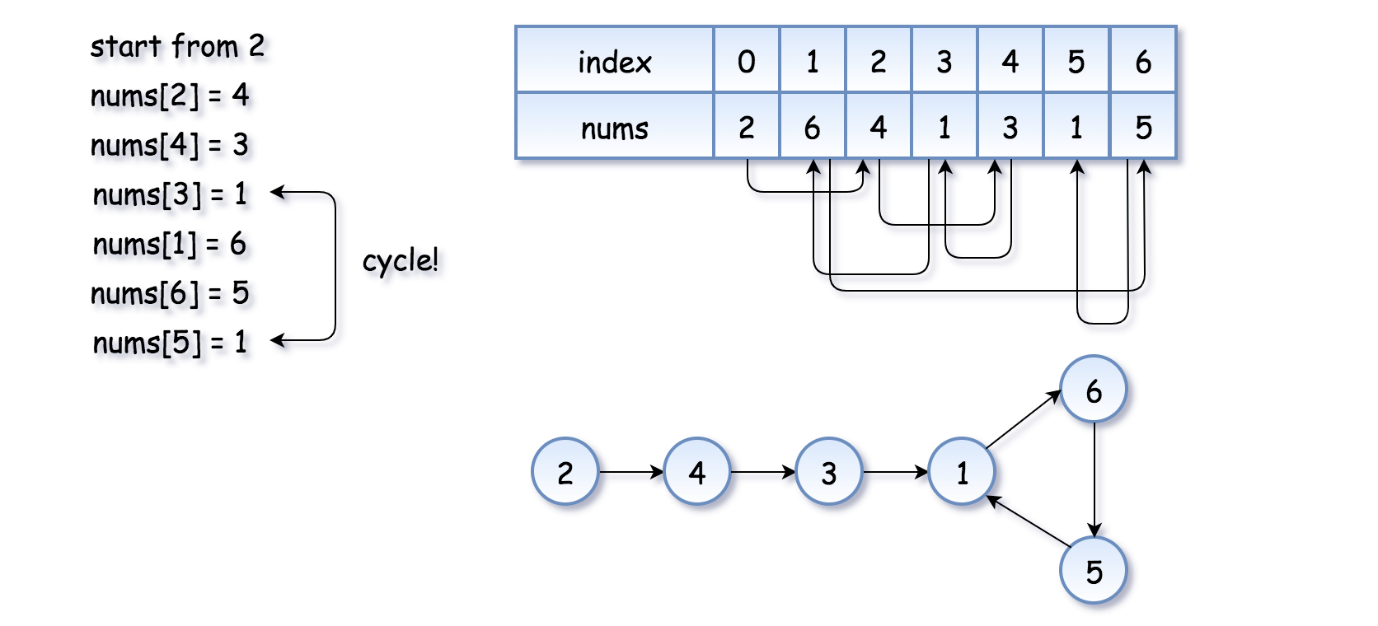
First of all, where does the cycle come from? Let's use the function f(x) = nums[x] to construct the sequence: x, nums[x], nums[nums[x]], nums[nums[nums[x]]], ....

Each new element in the sequence is an element in nums at the index of the *previous* element.

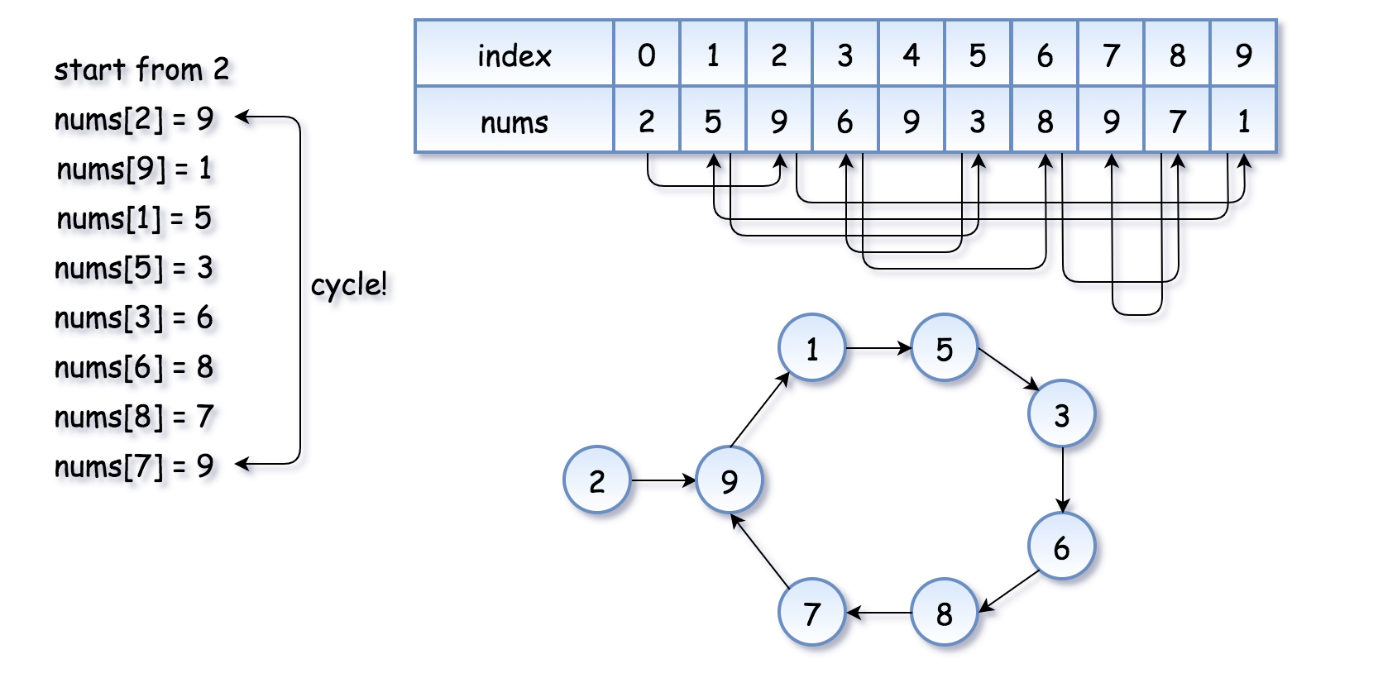
If one starts from x = nums[0], such a sequence will produce a linked list with a cycle.

The cycle appears because nums contains duplicates. The duplicate node is a cycle entrance.

Here is how it works:



The example above is simple because the loop is small. Here is a more interesting example (special thanks to @[sushant\_chaudhari](https://leetcode.com/sushant_chaudhari))



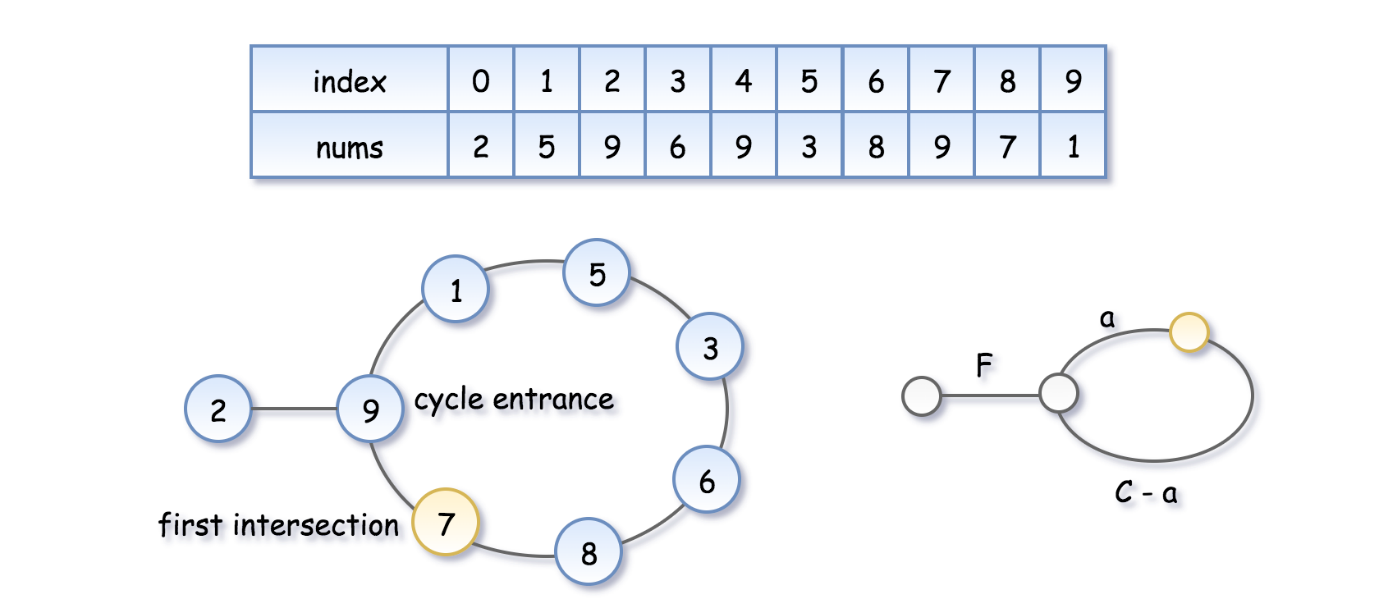
Now the problem is to find the entrance of the cycle.

**Algorithm**

[Floyd's algorithm](https://en.wikipedia.org/wiki/The_Tortoise_and_the_Hare) consists of two phases and uses two pointers, usually called tortoise and hare.

**In phase 1**, hare = nums[nums[hare]] is twice as fast as tortoise = nums[tortoise]. Since the hare goes fast, it would be the first one who enters the cycle and starts to run around the cycle. At some point, the tortoise enters the cycle as well, and since it's moving slower the hare catches the tortoise up at some *intersection* point. Now phase 1 is over, and the tortoise has lost.

Note that the intersection point is not the cycle entrance in the general case.

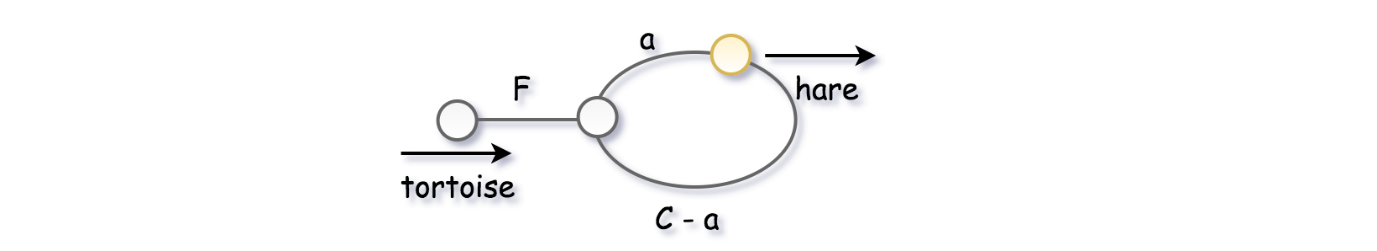


To compute the intersection point, let's note that the hare has traversed twice as many nodes as the tortoise, *i.e.* 2d(\text{tortoise}) = d(\text{hare})2*d*(tortoise)=*d*(hare), that means

2(F + a) = F + nC + a2(*F*+*a*)=*F*+*nC*+*a*, where n*n* is some integer.

Hence the coordinate of the intersection point is F + a = nC*F*+*a*=*nC*.

**In phase 2**, we give the tortoise a second chance by slowing down the hare, so that it now moves with the speed of tortoise: tortoise = nums[tortoise], hare = nums[hare]. The tortoise is back at the starting position, and the hare starts from the intersection point.



Let's show that this time they meet at the cycle entrance after F*F* steps.

* The tortoise started from zero, so its position after F*F* steps is F*F*.
* The hare started at the intersection point F + a = nC*F*+*a*=*nC*, so its position after F steps is nC + F*nC*+*F*, that is the same point as F*F*.
* So the tortoise and the (slowed down) hare will meet at the entrance of the cycle.