class Solution:

def firstMissingPositive(self, nums: List[int]) -> int:

n = len(nums)

# Phase 1: Place numbers in their correct positions (if possible)

# A number 'x' should ideally be at index 'x-1'.

i = 0

while i < n:

correct\_pos = nums[i] - 1

# Check if the number is positive, within the array bounds,

# and not already in its correct position.

if 1 <= nums[i] <= n and nums[i] != nums[correct\_pos]:

nums[i], nums[correct\_pos] = nums[correct\_pos], nums[i] # Swap

else:

i += 1 # Move to the next element

# Phase 2: Find the first missing positive

# Iterate through the modified array and find the first index 'i'

# where nums[i] is not equal to 'i + 1'.

for i in range(n):

if nums[i] != i + 1:

return i + 1 # This is the first missing positive

# If all numbers from 1 to 'n' are present, then 'n + 1' is the first missing positive.

return n + 1