Problem 163: Find Minimum in Rotated Sorted Array

<https://leetcode.com/problems/find-minimum-in-rotated-sorted-array/>

class Solution:

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

n = len(nums)

if n == 0:

return(nums)

elif n == 1:

return(nums[0])

low = 0

high = n - 1

if nums[low] < nums[high]:

return(nums[low])

while (low < high):

middle = (low + high)//2

if nums[middle] > nums[middle + 1]:

return(nums[middle + 1])

elif nums[middle] < nums[middle - 1]:

return(nums[middle])

else:

if nums[middle] > nums[low]:

low = middle

else:

high = middle

return(nums[low])