def permute(nums):

solutions = []

len\_nums = len(nums)

def fill\_blanks(idx\_subproblem):

# Base case

if idx\_subproblem == len\_nums:

#copy\_list = new List(nums)

solutions.append(list(nums)) # create copy of list

return

# Recursive case

# for (int i = 0; i < len\_nums; i++)

for idx\_pick in range(idx\_subproblem, len\_nums): # [0, len\_nums)

# swap

nums[idx\_pick], nums[idx\_subproblem] = nums[idx\_subproblem], nums[idx\_pick]

fill\_blanks(idx\_subproblem + 1)

# swap back

nums[idx\_pick], nums[idx\_subproblem] = nums[idx\_subproblem], nums[idx\_pick]

fill\_blanks(0)

return solutions