想法：從頭開走，紀錄最大能到的位子若超過終點則結束。如果中間卡了0無法往後就結束因此迴圈i是停在最大能到的位子 而不是list長度終點

class Solution:

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

n = len(nums)

maxPos = 0

i = 0

while i <= maxPos:

maxPos = max(maxPos, i + nums[i])

if maxPos >= n-1: //找到路到終點就return true

return True

i += 1

return False

想法dp：從頭開始往後

dp(0)往後找是否能到dp(n-1)的路線(return true)

class Solution:

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

n = len(nums)

def dp(i):

if i == n - 1:

return True

for j in range(i+1, min(i+nums[i], n-1) + 1): //若踩到0就會false表示此路不通

if dp(j):

return True

return False

return dp(0)