想法：用dp需要找到規律第一個做法從dp0開始往後但有時會timeout

第二個做法有機會跳步走只挑需要用到的計算

class Solution:

def numSquares(self, n: int) -> int:

dp = [0]

for i in range(1,n+1):

min\_num = float('inf')

j = 1

while i - j\*\*2 >= 0:

min\_num = min(min\_num, dp[i - j\*\*2] + 1)

j += 1

dp.append(min\_num)

return dp[n]

class Solution:

def numSquares(self, n: int) -> int:

dp = [0] + [float('inf')]\*n

for i in range(1, n+1):

dp[i] = min(dp[i-j\*j] for j in range(1, int(i\*\*0.5)+1)) + 1

return dp[n]