A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer program

AI-generated content may be incorrect. Dp:  
 1. Handle the base case  
 2. Create DP table  
 3. Fill up the table(Loop)  
 4. Return res  
   
 Sliding Window:  
 1. First window  
 2. Slide the window(loop  
 3. Return

Backtracking:  
 1. Some condition know add

If some condition:

Result.append(path)  
 return  
 2. Make a choice (Choose)  
 path.append(choice)

3. Explore

Backtrack(path,updated\_choice)

4. Undo the choice

Path.pop()

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

LeetCode 215 – Kth Largest Element in an Array : Input: nums = [3,2,1,5,6,4], k = 2 Output: 5

def findKthLargest(nums, k):

min\_heap = []

# Step 1: Add → Push all numbers into min-heap (keep only k elements)

for num in nums:

heapq.heappush(min\_heap, num)

# Step 2: Maintain → If heap size > k, remove smallest

if len(min\_heap) > k:

heapq.heappop(min\_heap)

# Step 3: Extract → Heap has k largest, smallest of them = kth largest

return min\_heap[0]