Tính độ phức tạp của các thuật sau (yêu cầu xác định từng bước)  
Bài 1:  
def binary\_search(arr, target):  
    left, right = 0, len(arr) - 1  
    while left <= right:  
        mid = (left + right) // 2  
        if arr[mid] == target:  
            return mid  
        elif arr[mid] < target:  
            left = mid + 1  
        else:  
            right = mid - 1  
    return -1  
  
  
Bai 2:    
def max\_subarray\_sum(arr):  
    max\_sum = arr[0]  
    current\_sum = arr[0]  
    for i in range(1, len(arr)):  
        current\_sum = max(arr[i], current\_sum + arr[i])  
        max\_sum = max(max\_sum, current\_sum)  
    return max\_sum

Giải:  
Bài 1: Độ phức tạp là O(logn). Vì binary\_search sẽ cắt từng đoạn xuống.

Bài 2: Độ phức tạp là O(n) vì chạy từ 1 đến hết mảng.