

Max Subarray Sum using Prefix Sum

```
# < Prefix Sum Approach to Find Max Subarray Sum
# Time Complexity: O(n²)
# -----

arr = [2, 4, -1, 3]
n = len(arr)

# Step 1: Build prefix sum array
prefix = [0] * n
prefix[0] = arr[0]

for i in range(1, n):
    prefix[i] = prefix[i - 1] + arr[i]

# Step 2: Try all subarrays using prefix sum
max_sum = float('-inf')

for i in range(n):
    for j in range(i, n):
        # Subarray sum using prefix
        if i == 0:
            sub_sum = prefix[j]
        else:
            sub_sum = prefix[j] - prefix[i - 1]

        print(f"Subarray ({i}, {j}) → Sum: {sub_sum}")

        if sub_sum > max_sum:
            max_sum = sub_sum

print("\n Maximum Subarray Sum:", max_sum)

Subarray (0, 0) → Sum: 2
Subarray (0, 1) → Sum: 6
Subarray (0, 2) → Sum: 5
Subarray (0, 3) → Sum: 8
Subarray (1, 1) → Sum: 4
Subarray (1, 2) → Sum: 3
Subarray (1, 3) → Sum: 6
Subarray (2, 2) → Sum: -1
Subarray (2, 3) → Sum: 2
Subarray (3, 3) → Sum: 3

 Maximum Subarray Sum: 8
```