

You

Write me a python code that solves the problem with such description: The maximum sum subarray problem consists in finding the maximum sum of a contiguous subsequence in an array or list of integers:

`max_sequence([-2, 1, -3, 4, -1, 2, 1, -5, 4])`

should be 6: [4, -1, 2, 1]

Copilot

Sure, here is the Python code that solves the maximum sum subarray problem:

```
def max_sequence(arr):
    # Initialize max_so_far and max_ending_here
    max_so_far = max_ending_here = 0

    # Initialize start and end variables
    start = end = s = 0

    for i in range(len(arr)):
        max_ending_here += arr[i]

        if max_so_far < max_ending_here:
            max_so_far = max_ending_here
            start = s
            end = i

        if max_ending_here < 0:
            max_ending_here = 0
            s = i + 1

    # Return the maximum sum and the subarray
    return max_so_far, arr[start:end+1]
```

When you run the function `max_sequence([-2, 1, -3, 4, -1, 2, 1, -5, 4])`, it returns (6, [4, -1, 2, 1]) which means the maximum sum of a contiguous subsequence in the array is 6 and the subsequence is [4, -1, 2, 1].

You

the function should only return int values

Copilot

Sure, I modified the function to return only the maximum sum. Here is the updated Python code:

```
def max_sequence(arr):
    # Initialize max_so_far and max_ending_here
    max_so_far = max_ending_here = 0

    for i in range(len(arr)):
        max_ending_here += arr[i]

        if max_so_far < max_ending_here:
            max_so_far = max_ending_here

        if max_ending_here < 0:
            max_ending_here = 0

    # Return the maximum sum
    return max_so_far
```

When you run the function `max_sequence([-2, 1, -3, 4, -1, 2, 1, -5, 4])`, it returns 6 which means the maximum sum of a contiguous subsequence in the array is 6.