Maximum Subarray Sum Problem

AmeerTaweel

September 15, 2022

Contents

1 Description

Given an array of n numbers, calculate the maximum subarray sum, which is the largest possible sum of a sequence of consecutive values in the array. There may be negative numbers in the array. Zero-length subarrays are allowed, so the maximum subarray sum is always at least zero.

2 Input

The first line of input contains an integer n, the number of elements in the array.

Then n lines follow, each containing a number.

```cpp
int n;
cin >> n;

vector<int> arr(n);
for (int i = 0; i < n; i++) cin >> arr[i];
```

Listing 1: Read Input

3 Output

One number, the maximum subarray sum.

```cpp
cout << max_sum;
```

Listing 2: Write Output

4 Solutions

4.1 Three Loops

4.1.1 Algorithm

This algorithm coputes the sum for all sequences and then outputs the maximum.

```
int max_sum = 0;

for (int start = 0; start < n; start++) {
    for (int end = start; end < n; end++) {
        int sum = 0;
        // Calculate sum for sequence [start, end]
        for (int i = start; i <= end; i++) sum += arr[i];
        if (sum > max_sum) max_sum = sum;
    }
}
```

Listing 3: Three Loops Algorithm

4.1.2 Time Complexity

The algorithm above has three nested loops that iterate through the output, so its time complexity is $O(n^3)$.

4.2 Two Loops

4.2.1 Algorithm

This algorithm improves on the previous one by calculating the sum as the end pointer moves. Instead of doing the whole calculation at each step.

```
int max_sum = 0;

for (int start = 0; start < n; start++) {
    int sum = 0;
    for (int end = start; end < n; end++) {
        sum += arr[end];
        if (sum > max_sum) max_sum = sum;
    }
}
```

Listing 4: Two Loops Algorithm

4.2.2 Time Complexity

The algorithm above has two nested loops that iterate through the output, so its time complexity is $O(n^2)$.

4.3 One Loop - Kadane's Algorithm

4.3.1 Algorithm

```c
int max_sum = 0;

int sum = 0;
for (int i = 0; i < n; i++) {
    if (sum < 0) sum = 0;
    sum += arr[i];
    if (sum > max_sum) max_sum = sum;
}
```

Listing 5: Kadane's Algorithm

4.3.2 Time Complexity

The algorithm above has only one loop that iterate through the output, so its time complexity is $O(n)$.