Kadane algo

28 June 2023 20:27

Given an integer array nums, find the subarray with the largest sum, and return its sum.

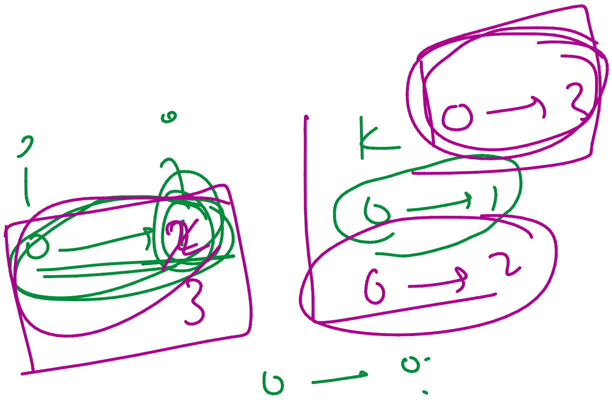
[-2,1,-3,4,-1,2,1,-5,4]

Subarray → contiguous part of an array

[1,2,3,4] → [1], [2], [3], [4], [1,2], [1,2,3], [1,2,3,4], [2,3], [2,3,4], [3,4]

[1,2,3,4] → 10

[-1,-2,-3,-4] → -1



loop [-2,1,-3,4,-1,2,1,-5,4]

[-2], [-2,1], [-2,1,-3], [-2,1,-3,4], [-2,1,-3,4,-1], [-2,1,-3,4,-1,2], [-2,1,-3,4,-1,2,1], [-2,1,-3,4,-1,2,1,-5], [-2,1,-3,4,-1,2,1,-5,4]

[1], [1,-3], [1,-3,4], [1,-3,4,-1], [1,-3,4,-1,2], [1,-3,4,-1,2,1], [1,-3,4,-1,2,1,-5], [1,-3,4,-1,2,1,-5,4]

[-3], [-3,4], [-3,4,-1], [-3,4,-1,2], [-3,4,-1,2,1], [-3,4,-1,2,1,-5,4] ...

Ans [4,-1,2,1] → 6

ans = INT_MIN → -∞ → Most -ive value in CPP

```
for (int i=0; i<n; i++)
{ int sum = 0;
  for (int j=i; j<n; j++)
  {
    sum += arr[j];
    ans = max(sum, ans);
```

9 → 16 s

10000 → 18000

10^7

10000 → 18000

1 sec → 10^8 computations

10^7

10^8

18×10^7

10 → 55 times

---

sum = 0

-2,1,-3,4,-1,2,1,-5,4

[-2], [-2,1], [-2,1,-3], [-2,1,-3,4]

1 [1], [1,-3], [1,-3,4]

---

i	ans	cursum
0	-∞	0

[-2,1,-3,4,-1,2,1,-5,4]

```
for (int i=0; i<n; i++)
    cursum += arr[i]
    ans = max(cursum, ans)
    if cursum < 0
        (cursum = 0)
```

[1, -3] = -2

5 time >> 5 time -2