Dp MCM 9

Dynamic Programming

2 Player game, intuitively, a player turn wi on pick from one of the ends, Person with max sum wins

arr[:] : 3 5 8 5 7 3 4 5 | arr[:] : 3 5 8 5 7 3 4 5 | Values: 8 19 19 22 | Values: 8 7 10 19 22

arr : 3 5 8 20 19 6 1 12 7 3

dp[i,j] : Max sum we can obtain from [i:j] as a player turn

Player 1 starts by pick max sum

$\{0-4\}$ : Player 1 pick max sum

$\{1-4\}$ : Player 2 left by to max left sum

$\{0-3\}$ : Player 2 left by to max right sum

Player 1 zone

Player 2 zone

Player 1 zone

Player 2 zone

Max sum

$dp[0,n-1] - dp[1,n-1]$

$dp[0,n-1] - dp[0,n-2]$

$dp[0,n-1] - dp[1,n-1]$

$dp[0,n-1] - dp[0,n-2]$

$\{0-3\}$ Player 1 turn

$dp[i,j] = \max\begin{cases} arr[i] + sum[i+1,j] - dp[i+1,j] \\ arr[j] + sum[i,j-1] - dp[i,j-1] \end{cases}$

dp expression :

$dp[i,j] = \begin{cases} \{i=j\} : \{\text{Single element}\} \; dp[i,i] = arr[i] \\ \{i=j-1\} : \max(arr[i], arr[j]) \\ \max\begin{cases} arr[i] + sum[i+1,j] - dp[i+1,j] & \text{Player 1 can get} \\ \text{using prefix sum} & i, j-1 \\ arr[j] + sum[i,j-1] - dp[i,j-1] & \text{Player 1 can get} \end{cases} \end{cases}$

Player 1 Max sum : $dp[0,N-1]$

Player sum : $sum[0,N-1] - dp[0,N-1]$

: Which sum is more than previous player left up.

$\{1,2,3\}$ Player left choosing into 2

$\{1,2,3\}$ Player 1 left choosing into 2

$\{0\}$ if $\{1\}$ sum Player left turn?

Player 1 : 1

Player 2 : 100

Player 1 : 5 10

Player 2 : 100 5

Prob : [1,2,3]

dp[i][N]

TC : # of dp state * TC of dp state

$N(N) + O(1)$, dp gets can using prefix array

2d3 Return an array per non overlapping zone

2d3 Return an 2D-Array, per non overlapping zone (max)

Simple problem

2D-Array : M

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | -3 | 8 | -2 | -1 | 10 |
| 1 | 5 | 2 | -12 | 5 | 7 |
| 2 | 3 | 2 | 2 | -1 | -10 |
| 3 | -1 | 6 | -3 | 4 | 1 |
| 4 | 7 | 10 | -3 | 2 | -1 |

N

Zone : 2 5

How : 1) For all subarrays, get sum

How to find subarrays ?

1) Top left + bottom right by $\{i,j\}$ to picking a point to create

$O(n^2)(n,m)$

$= N^2 m_{C_2} = O(N^2)(M^2)$

For each subarray to get sum

Zone : Unique $arr[i,j]$

Prefix sum, $O(1)$ on sub array

Solution :

1) $O(n^2)(m^2) + O(1)$

2) $O(n^2)(m^2) + O(1)$ : Normally do subarray + find a subarray sum using prefix sum

$P[i][j] - \text{Subarray sum from } (1,1) \text{ to } (i,j)$

2) Find max subarray sum where non overlapping

Should sum at $\boxed{0}$ & at $\boxed{1}$ =

Max subarray sum.

$\boxed{0}$ $\boxed{10}$ $\boxed{-15}$ $\boxed{1}$ $\boxed{1}$ $\boxed{1