

```
Exp-2C
Max - Subarray & Sum
```

$$*$$
 Array = 1, -3, 2, 1, -1

1) Brute

if this Sum 7 max Sum

computes sum for all possible subarrays & updates max Sum O(n2); simple but slow

 $[1] \Rightarrow 1 \qquad , [-3,2] \Rightarrow -1, [-3,2,1] \Rightarrow 0, [2,1] \Rightarrow 3$ $\begin{bmatrix} 1, -3 \end{bmatrix} \Rightarrow -2 , \begin{bmatrix} -3, 2, 1, -1 \end{bmatrix} \Rightarrow -1, \begin{bmatrix} 2 \end{bmatrix} \Rightarrow 2, \begin{bmatrix} 2, 1, -1 \end{bmatrix} \Rightarrow 2$

 $[1,-3,2] \rightarrow 0$, $[1] \rightarrow 1$, $[1,-1] \rightarrow 0$

 $[1,1-3,2,1] \Rightarrow 1$ $\begin{bmatrix} -3 \end{bmatrix} \Rightarrow -3$

Max Sum => 3

2) Divide & Conquer

o(nlogn); recurgive

$$[1, -3]$$
 $[2, 1, -1]$

Trace recursion:

$$\begin{bmatrix} 1 \end{bmatrix} \Rightarrow 1$$
, $\begin{bmatrix} 2 \end{bmatrix} \Rightarrow 2$
 $\begin{bmatrix} -3 \end{bmatrix} \Rightarrow 3$, $\begin{bmatrix} 1 \end{bmatrix} \Rightarrow 1$, $\begin{bmatrix} -1 \end{bmatrix} \Rightarrow -1$

3.7 Kadane's

 \rightarrow O(n); most eff.

i=0: max Sum =1

 $i=1: thisSum = -2, m_S = 1 \Rightarrow thisSum = 0$

i = 2: this Sum = 2, $m_{-}S = 2$ i = 3! this Sum = 3, $m_{-}S = 3$ = 7 [2, 1] i = 4: this Sun = 2, $m_{-}S = 3$