Max Sum sub matrix

1. Full Search (O(n^6)

a[1][1] a[1][2] …… a[1][j1] …… a[1][j2] … a[1][m]

a[2][1] a[2][2] …… a[2][j1] …… a[2][j2] … a[2][m]

……

a[i1][1] a[i1][2] … a[i1][j1] … a[i1][j2] … a[i1][m]

…… …… ……

…… …… ……

…… …… ……

a[i2][1] a[i2][2] … a[i2][j1] … a[i2][j2] … a[i2][m]

……

……

a[n][1] a[n][2] … a[n][j1] … a[n][j2] … a[n][m]

2,	10,	8,	3,
-8,	14,	-1,	4,
-6,	-1,	8,	-2,
1,	8,	7,	3,
8,	2,	-10,	-8,

10	8	3
14	-1	4
-1	8	-2
8	7	3

2. Help matrix (O(n^4))

2.1 Build HELP matrix

mat[][] = || a[i][j] ||

i = 0 – n j = 0 – m

help[i][j] :

1. help[0][0] = mat[0][0]

2. help[i][0] = help[i-1][0] + mat[i][0]

3. help[0][j] = help[0][j-1] + mat[0][j]

4. help[i][j] = mat[i][0] + help[i-1][j]

+ help[i][j-1] - help[i-1][j-1]

```
static int[][] helpMatrix(int[][]mat){//O(N^2)
```

2.2 Calculate Sum ij_pq

$$
\text{Sum} = \begin{cases}
\text{help}[p][q] & \text{if } j == 0 \;\&\; j == 0 \\
\text{help}[p][q] - \text{help}[p][j-1] & \text{if } j == 0 \;\&\; j > 0 \\
\text{help}[p][q] - \text{help}[i-1][q] & \text{if } j > 0 \;\&\; j == 0 \\
\text{help}[p][q] - \text{help}[p][j-1] - \text{help}[i-1][q] + \text{help}[i-1][j-1] &
\end{cases}
$$

```
static int sum_ij_pq(int help[][], int i, int j, int p, int q){//O(1)
```

2.3 Calculate Max Sum

```
for i
    for j
        for p
            for q
```

2. HELP matrix

```
 2,  10,    8,   3,
-8,   14,  -1,   4,
-6,   -1,   8,  -2,
 1,    8,   7,   3,
 8,    2, -10,  -8,
```

first column

2	**2**
-8 +2 = -6	**-6**
-6 – 6 = -12	**-12**
1 – 12 = -11	**-11**
8 – 11 = -3	**-3**

//first row

2 10 + 2 =12 8 + 12 = 20 3 + 20 = 23

2	**12**	**20**	**23**
-6			
-12			
-11			
-3			

help[i][j] = mat[i][j] + help[i-1][j] + help[i][j-1] - help[i-1][j-1]

2	12	20	23
-6	18		
-12	11		
-11	20		
-3	30		

14 + 12 -6 – 2 = 18

-1 +18 – 12 + 6 = 11

8 + 11 – 11 +12 = 20

2 + 20 – 3 + 11 = 30

2,	10,	8,	3,
-8,	14,	-1,	4,
-6,	-1,	8,	-2,
1,	8,	7,	3,
8,	2,	-10,	-8,