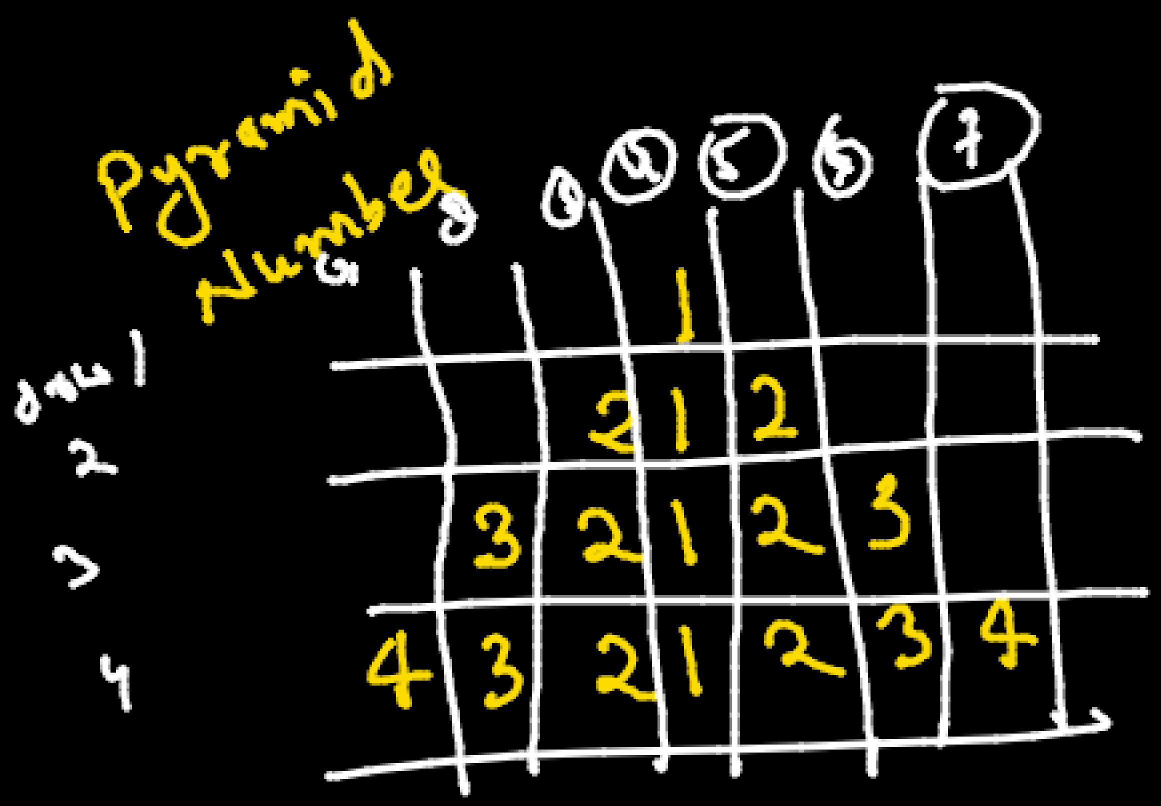
2nd largest

4 4 5 5

```
sort (arr, arr+n);
maxi = arr[n-1];
for(int i=n-2; i>=0; i--)
{ if(maxi > arr[i])
  { cout << arr[i];
    return 0;
  }
}
```



row → 1 to 4

col 1 to 7

row	col (pos)
1	4
2	3 4 5
3	2 3 4 5 6
4	1 2 3 4 5 6 7

if (col >= n-row+1 && col <= n+row-1) { print here; }

n = 4

n - row + 1 && n + row - 1

~~4-1+1~~ = 4   4+1-1 = 4

4-2+1 = 3   4+2+1 = 5

4-3+1 = 2   4+3-1 = 6

4-4+1 = 1

```
    1
  2 1 2
3 2 1 2 3
4 3 2 1 2 3 4
```

```
for (row=1 ; row<=n ; row++)
{   int val=row;
    for (col=1 ; col<=(2*n)-1; col++)
    {
        if (col >= n-row+1 && col<=n+row-1)
        {
            cout<<val;
            if( col>=n-row+1 && col<n)
                val--;
            else
                val++;
        }
        else{
            cout<<" ";
```

1 1 2 3 4 5 5 4 3 2 1
1 2 3 4 * * 4 3 2 1
1 2 3 * * * * 3 2 1
1 2 * * * * * * 2 1
1 * * * * * * * * 1

row = 1 to 5
col : 1 to 10

n row col
5

```
row → 1 to 5
  col → 1 to 2n
                    int val1 = 1
                        val2 = n - row + 1
  if ( col ≤ n - row + 1 )
      cout << val1++;
  else if ( col ≥ n + row )
      cout << val2--;
  else cout << "*";
```

row≥1
row
1

5 n - row + 1 && n + row
5 - 1 + 1 = 5
5 - 2 + 1 = 4
5 - 3 + 1 = 3
5 - 4 + 1 = 2
5 - 5 + 1 = 1

print term 1

5 + 1
5 + 2
5 + 3
5 + 4
5 + 5

print term 2