optimization to make Ackermann more effiden

⇒ to solve this problem, we use Dynamic programm

First: create 2d Array of size [m+1][n+1
that will hold the values of Ack(0,0) unt
Ack(m,n).

second: Filling the first row of 2d Array us
the first case Ack = n+1 as m = 0

m \ n	0	1	2	3
0	1	2	3	4
1	2	3	4	5
2	3	5	7	
3	5			

Space Complexity O(m×n).

ime complexity O(m×n).

2019oo50 محمد عصام الدين عبد الفتاح ابراهيم

p.2 Ackermann Funct.

- why base Ackermann is so heavy to compute.

⇒ For recursive solution that terminates with va
greater than (4,1), in each recursive ca
either "m" decrease or "m" remains thes
and "n" decrease.
So it has O(m) space complexity.
and O(m ack(m,n)) time complexity.

when "m" decrease there is no idea about
e maximum value "n" will increase.

Kermann function will grow relatively ~~with~~ when
n={1,1,2,3} but when m>=4, it will grow
astly in unobserved order.