

Hey Ackermann!

ZPRAC-16-17-Lab9

Hey Ackermann [30 points]

ANNOUNCEMENT:

Up to 20% marks will be allotted for good programming practice. These include

- Comments for non trivial code
- Indentation: align your code properly

Up to 50% marks can be deducted if you do not use recursion

Use long long int data type to store integers for this problem. Do not worry about overflow issues.

The **Ackermann** function is a well known recursive function defined for $m \geq 0, n \geq 0$ as follows:

$A(m, n) =$

- $n+1$, if $m=0$
- $A(m-1, 1)$, if $m>0$ and $n=0$
- $A(m-1, A(m, n-1))$, if $m>0$ and $n>0$

Find **ackermann**(m, n) for the given m and n . The ackermann function should be defined using recursion only (hence use of arrays is not allowed).

Also print the number of recursive calls made to the function (Note: Use only the above formulation). Each call to the function Ackermann() should be considered as a separate call.

NOTE: The ackermann function should be defined using recursion only (hence use of arrays is not allowed). Follow the exact recurrence given above, otherwise you may face issues in the count-of-recursive-calls.

Input Format:

The first line of input is a number t which indicates the number of test cases. Then, t lines follow where each line contains the indices m and n separated by space. $1 \leq t, m, n \leq 5$

Output Format:

For every test case, output two integers, separated by a space

(i) the term of the series corresponding to indices m and n

(ii) the number of recursive calls to ackermann function.