Mathematical Analysis for Recursive alg:

General plan/ pradure for analysing efficiency of Recursive alg:-
↳ Function calling itself

① Decide the I/p size based on parameter

② Identify the alg's basic operations.

③ Check how many times the basic operation is executed.

↳ Find the whether execution of basic operation depends upon I/p size.

↳ Determine best, Average, worst case for I/p size 'n'.

④ Set up the recurence relation with some initial condition & expressing the basic operation

⑤ Used forward / Backward Substistances method.

Ex:1 Explain the recursive alg for computing Factorial & analyze alg.

Problem Statement:

Computing factorial of a no using recursion.

Alg:

Alg factorial (n).

// problem description: This alg computes n! using recursive fun

///P: A non-negative integer 'n'.

//O/P: returns the factorial value.

```
if (n=0)
return 1
else
    return factorial (n-1)*n.
```

Ex:

If factorial of a number can be obtained by performing repeated multiplication.

If n=5, then. $5! = 5 \times 4 \times 3 \times 2 \times 1$.

Step1: n! = 5!

Step2 = 4! * 5

Step3 = 3! * 4 * 5

Step4 = 2! * 3 * 4 * 5

Step5 = 1! * 2 * 3 * 4 * 5.

Step6 = 0! * 1 * 2 * 3 * 4 * 5 (0! = 1)

Step7 = 1 * 1 * 2 * 3 * 4 * 5

O/P = n! = 5! = 120.

n! => General term:

$n! = n \times (n-1) \times (n-2) \times (n-3) \times (n-4) \times \ldots$

$5! = 5 \times 4!$

$F(n) = F(n-1) * n$