**KARNATAK LAW SOCIETY’S**

**GOGTE INSTITUTE OF TECHNOLOGY**

**UDYAMBAG, BELAGAVI – 590008**

**(An Autonomous Institution under Visvesvaraya Technological University, Belagavi)**

**(Approved By AICTE, New Delhi)**

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

 

**COURSE PROJECT**

**DESIGN AND ANALYSIS OF ALGORITHM**

RACHANA KAMPLI 2GI18IS032

LAXMI NYAMAGOUD 2GI18IS020

HEMANTH I T 2GI18IS015

ROHAN KOKATANUR 2GI18IS066

Guided by:

**Prof. Gururaj Kulkarni**

**TITLE: RECURRENCE RELATION**

**OBJECTIVES:**

TO LEARN

* WHAT IS RECURRENCE RELATION
* METHODS FOR SOLVING RECURRENCE RELATIONS
* COMMON RECURRENCE TYPES IN ALGORITHM ANALYSIS
* CALCULATION OF TIME EFFICIENCY

**DEFINITION:**

* In Mathematics, a RECURRENCE RELATION is an equation that recursively defines a sequence or multidimensional array of values, once one or more initial terms are given; each further term of the sequence or array is defined as a function of the preceding terms.
  + - x(n)=x(n-1) +n for n >0 is called a recurrence relation or recurrence equation
    - x (0) =0 is called initial condition

**METHODS FOR SOLVING RECURRENCE RELATIONS**

* Method of forward substitution
* Method of backward substitution
* Linear second-order recurrences with constants coefficients
* Mathematical analysis of recursive algorithm

**COMMON RECURRENCE TYPES IN ALGORITHM ANALYSIS**

* Decrease by one
* Decrease by constant factor
* Divide and conquer

**MATHEMATICAL ANALYSIS OF RECURSIVE ALGORITHM**

**ALGORITHM TO CALCULATE ‘N’ FACTORIAL**

* //Input: - a non-negative integer n
* //output: - the value of n!

if n=0 return 1

else return f(n-1) \*n

**Calculating time efficiency:**

Using Back Substitution Method

M(n)=M(n-1) +1

For n=0, i=n

M(n-1) =[M(n-2) +1]

M(n)=[M(n-2) +1] +1

M(n)= M(n-1) +1

M(n)= M(n-n) +n

M(n)=M (0) +n

M(n)=1+n

The basic operation is “MULTIPLICATION”

In this algorithm the basic operation gets executed ‘n-1’ times

The time efficiency of algorithm is theta(n)

**Conclusion:**

We get to know what is recurrence relation, methods for solving a recurrence relation and recurrence type in algorithm analysis and we analyse a algorithm and find its time efficiency, basic operation and how many times the basic operation gets executed.