SNS COLLEGE OF TECHNOLOGY

Coimbatore-35.

An Autonomous Institution

COURSE NAME : 19ITB201 DESIGN AND ANALYSIS OF ALGORITHMS

II YEAR/ IV SEMESTER

UNIT – I INTRODUCTION

TOPIC - ANALYSIS FRAMEWORK

Fundamentals of Analysis and Framework

The Analysis Framework :

Two types of efficiency :

Time efficiency or Time complexity – how fast an algorithm runs

Space efficiency or Space complexity – space required for an algorithm to run

ANALYSIS FRAMEWORK

1.1 Measuring an input size
1.2 Units for measuring running time
1.3 Orders of growth
1.4 Worst-case, best-case and average-case efficiencies
1.5 Recapitulation of the analysis algorithm

ANALYSIS FRAMEWORK

1.1 Measuring an input size:

Most of the time, we calculate input size based on the number of items in the input. For example, for sorting n integers in an array, we have n data elements, so input size = n. We perform basic operations like comparison, swapping, etc., to sort the input.

ANALYSIS FRAMEWORK

1.2 Units for measuring running time :

Some standard unit of measurement such as second , millisecond etc. to measure the running time of a program.

Drawbacks:

Depends on speed of computer

Depends on the quality of a program

Difficulty of checking actual running time.

ANALYSIS FRAMEWORK

1.3 Orders of growth :

It depends on the nature of the function :

1. Logarithmic function
   - Grows slowly
   - Count basic operation for all realistic function
2. Exponential function
   - Grows fastly
   - Count basic operation for all the range of numbers
3. Linear function

ANALYSIS FRAMEWORK

1.4 Worst-case, best-case and average-case efficiencies :

```
ALGORITHM SequentialSearch(A[0..n − 1], K)
//Searches for a given value in a given array by sequential search
//Input: An array A[0..n − 1] and a search key K
//Output: The index of the first element in A that matches K
// or −1 if there are no matching elements
i ←0
while i < n and A[i] = K do
i ←i + 1
if i < n return i
else return −1
```

ANALYSIS FRAMEWORK

1.4 Worst-case, best-case and average-case efficiencies :

Worst-case efficiency :

- It is the longest time taken to cover all the instances of n for a given problem to produce desired results.

Best-case efficiency :

- It is the shortest time taken to cover all the instances of n for a given problem to produce desired results.

Average-case efficiency :

- It is the average time taken to cover all the instances of n for a given problem to produce desired results.

ANALYSIS FRAMEWORK

1.4 Worst-case, best-case and average-case efficiencies :

Amortized efficiency :

- It applies not to a single run of an algorithm but rather to a sequence of operations performed on the same data structure.

ANALYSIS FRAMEWORK

1.5 Recapitulation of the analysis algorithm :

Important analysis framework :

- Both time and space efficiencies are measured as functions of the algorithm’s input size.
- Time efficiency is measured by counting the number of times the algorithm’s basic operation is executed.
- Space efficiency is measured by counting the number of extra memory units consumed by the algorithm.
- algorithms may differ significantly for inputs of the same size - worst-case, average-case, and best-case efficiencies are used.