Analysis of Algorithms By Smita Sankhe Email id: smitasankhe@somaiya.edu





Algorithms and Programs

Algorithm: a method or a process followed to solve a problem. • A recipe.

An algorithm takes the input to a problem (function) and transforms it to the output. A mapping of input to output.

A problem can have many algorithms. Preparing tea..





What is an Algorithm?

An algorithm is a finite set of instructions that, if followed, accomplishes particular task. In addition, all algorithms must satisfy the following criteria:

- 1. Input. Zero or more quantities are externally supplied.
- 2. Output. At least one quantity is produced.
- **3. Definiteness**. Each instruction is clear and unambiguous.
- **4. Finiteness**. If we trace out the instructions of an algorithm, then for all cases, the algorithm terminates after a finite number of steps.
- **5. Effectiveness**. Every instruction must be very basic so that it can be carried out, it also must be feasible.





Analysis of Algorithms

- <u>Syllabus</u>
- <u>IA</u>
- Lab Rubrics
- All experiments are to be done individually
- Experiment Grading
- *Experiments will be graded on
 - \circ Originality of work
 - \circ Timeliness (coding + write-up)
 - \circ Quality of answer





Practical Examination

- Final practical exam list may include the programs not performed during practical session
- The questions will be based on entire Analysis of Algorithms course
- The Viva and Practical exam marks distribution will be intimated in due course of time





Algorithm Properties

An algorithm possesses the following properties: o It must be correct.

- It must be composed of a series of concrete steps.
- There can be no ambiguity as to which step will be performed next.
- It must be composed of a finite number of steps.
- o It must terminate.





Algorithm Efficiency

There are often many approaches (algorithms) to solve a problem. How do we choose between them?

At the heart of computer program design are two (sometimes conflicting) goals.

- 1. To design an algorithm that is easy to understand, code, debug.
- 2. To design an algorithm that makes efficient use of the computer's resources.





Analysis of Algorithm

- Apriori analysis
 - o Time
 - \circ Space
 - Cost (Software Engineering)
- Posterior analysis





A Priori analysis	A Posteriori Testing
	It is done after execution of an
It is done before execution of an	algorithm. Or after writing the
algorithm.	program
Priori analysis is an absolute analysis.	Posteriori analysis is a relative analysis.
It is independent of language of compiler and types of hardware/OS.	It is dependent on language of compiler and type of hardware/OS
It will give approximate answer.	It will give exact answer.
It uses the asymptotic notations to represent how much time the algorithm will take in order to	It doesn't use asymptotic notations to represent the time complexity of an
complete its execution.	algorithm.



- Buying a cellphone
 - o Budget
 - \circ User age group
 - o Technical specification
 - o User reviews





- Buying a home
 - o Budget
 - \circ Area (price to area ratio)
 - Location & Locality
 - \circ Amenities in the apartment
 - $\ensuremath{\circ}$ Amenities around the place
 - \circ other factors





• Preparing for examination

- \circ # of chapters
- \circ #of days/hours in hand
- \circ Weightage given to every topic
- o Difficulty level
- \circ Importance of examination score





- Admission for higher studies
 - 0?
 - 0?
 - 0?
 - 0?





How to analyze algorithms

- Time
- Space
- Performance Analysis
 - ✤ Best Case
 - Average Case
 - Worst Case





Specifications of good Algorithm

- Work Correctly for all case
- Steps are clear
- Effective Time utilization
- Effective Space utilization
- Give best solution





Algorithm Classification

- Recursion
- Divide and Conquer Technique
- Greedy Technique
- Dynamic Programming Technique
- Backtracking Technique
- String Matching Algorithms
- Non-deterministic Polynomial Algorithms





Writing an Algorithm

- Comments begin with // and continue until the end of line.
- Blocks are indicated with matching braces: { and }.
- A compound statement (i.e., a collection of simple statements) can be represented as a block. The body of a procedure also forms a block.
- Statements are delimited by ;
- An identifier begins with a letter.
- Assignment of values to variables is done using the assignment statement (variable) := (expression);
- There are two Boolean values true and false.
- The logical operators and, or, and not
- The relational operators <,>,<=,>=,== and !=
- Array indices start at zero.
- The following looping statements are employed: for, while, and repeat-until.
- Conditional statements are if, if else and case





Writing an Algorithm

- Algorithm consist of two parts
- Head and Body

Algorithm Name ((parameter list))

where Name is the name of the procedure and ((parameter list)) is. The body has one or more a listing of the procedure parameters.

Algorithm Max(A, n)



