UNIT 1: Introduction

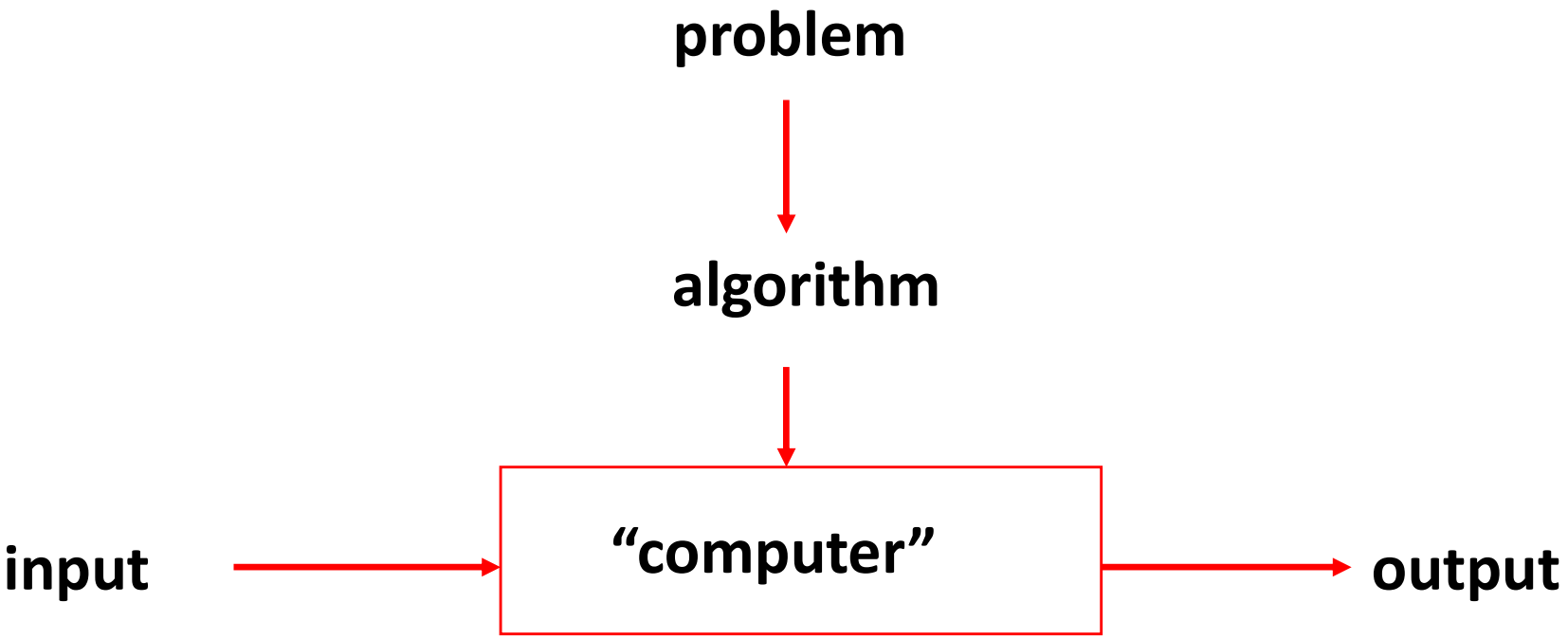
Notion of algorithm, Fundamentals of Algorithmic Problem Solving

- **Algorithm**
  - A sequence of unambiguous instructions for solving a problem, i.e., for obtaining a required output for any legitimate input in a finite amount of time.

- **Algorithmics**
  - study of algorithms

- **Why study Algorithmics?**
  - Indispensable computer applications that need algorithms
  - Developing analytical skills

Notion of algorithm



Algorithms – some important points

- Can be represented in various forms (pseudocode, English-like sentences, flowchart etc.,)
- Unambiguity/clearness
- Effectiveness
- Finiteness/termination
- Correctness

Characteristics of an algorithm

- Input: zero or more quantities are externally supplied
- Output: at least one quantity must be produced
- Definiteness: Each instruction is clear and unambiguous
- Finiteness: Algorithm must terminate after a finite number of steps
- Effectiveness: Every instruction must be basic i.e. simple instruction.

Methods of finding GCD

Competition

Computing Greatest Common Divisor: gcd(m,n)

Primary School

1. $t := \min(m, n)$
2. $m \bmod t = 0$?
3. Yes? $n \bmod t = 0$? Return t
4. No? $t = t\text{-}1$; goto 2

M - 1

Secondary School

1. Find prime factors of m
2. Find prime factors of n
3. Identify common factors
4. Return product of these

M - 2

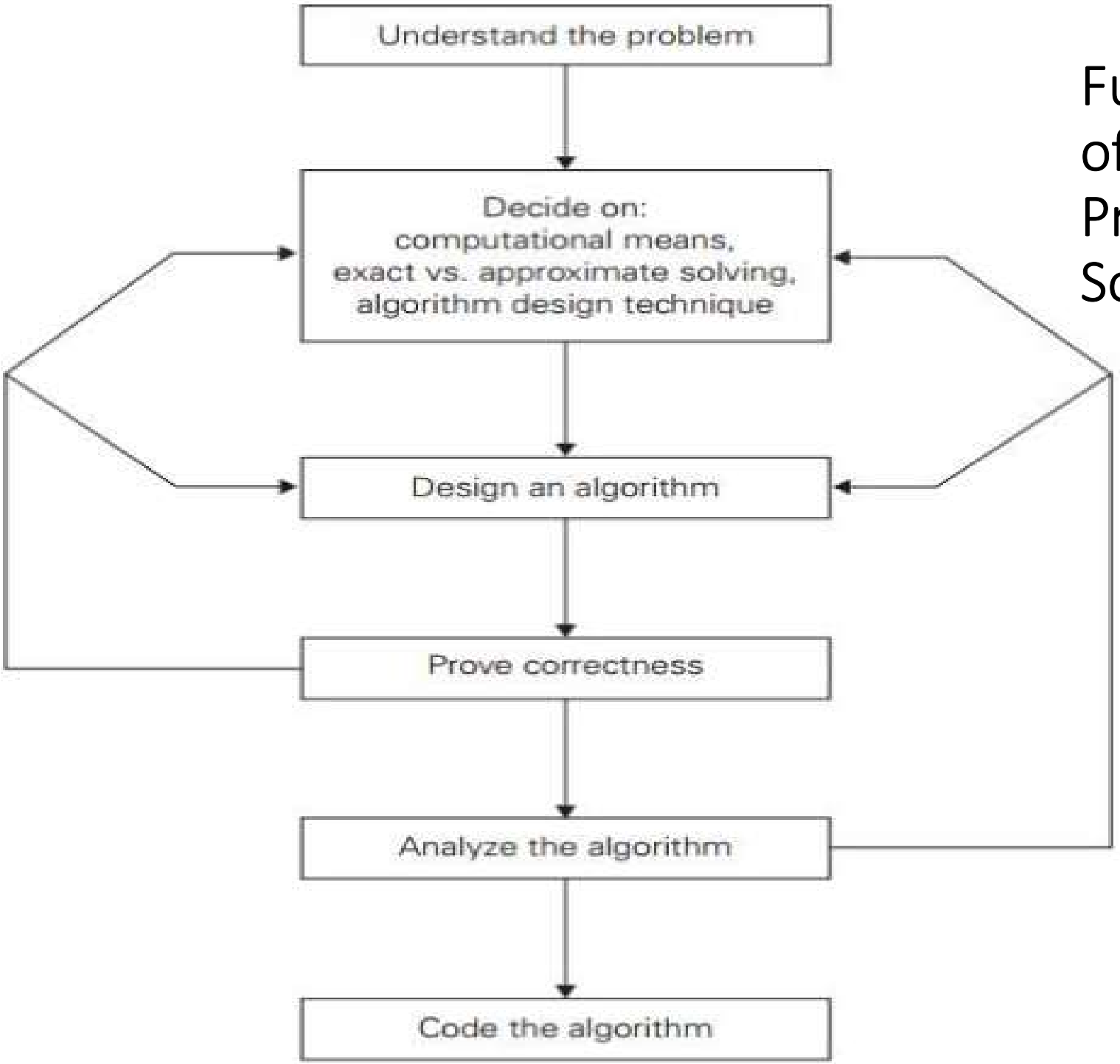
University

1. $n = 0$?
2. Yes? Return m
3. $r = m \bmod n$, $m := n$ $n := r$
4. Go to 1

M - 3

Fundamentals of algorithmic problem solving

Fundamentals of Algorithmic Problem Solving



Understanding the problem

- Understand the complete problem
  - solve problems by hand,
  - ask doubts,
  - think about special cases

- Instance:
  - range of inputs, valid/invalid inputs
  - boundary cases

Ascertaining the capabilities of a computational devices

- Sequential vs parallel algorithm
- Device limitations
- Fast vs slow computations
- Simple vs complex nature of volume of data / process / application

Choosing between exact and approximate problem solving

- Exact algorithm vs approximation algorithm (extracting square roots, solving non-linear equations etc.)
- Check for intrinsic complexity of the problem (e.g. TSP)
- Check if approximation algorithm is part of a more sophisticated algorithm that solves a problem exactly.

Deciding on appropriate data structure

- Structuring or re-structuring of data specifying problem's instance.

Algorithms + Data structures = Programs

Algorithm design techniques

- Technique / strategy / paradigm
- A general approach to solving problem algorithmically that is applicable to a variety of problems from different area of computing
- Classifies algorithms according to an underlying design idea. (categorize and study)

Methods of specifying an algorithm

- Natural language
- Pseudocode
- Flowchart

Proving an algorithm's correctness

- Mathematical induction – common technique
- Incorrect algorithm – Just one instance of input for which the algorithm fails.
- Approximation algorithm – show that error produced by an algorithm does not exceed predefined limit

Analyzing an algorithm

- Time efficiency : how fast the algorithm runs?
- Space efficiency : how much extra memory the algorithm needs?
- Simplicity : easier to understand and program
- Generality : range of inputs

Coding an algorithm

- Use programming language – OO concepts, Structural etc.

Important problem types

- Sorting
- Searching
- String processing
- Graph problems
- Combinatorial problems
- Geometric problems
- Numerical problems

Next session…

Fundamentals of the Analysis of Algorithmic Efficiency:

- Analysis framework,
- Asymptotic Notations and
- Basic Efficiency Classes

Puzzle time…

Torch and Bridge problem

There are 4 people (A, B, C and D) who want to cross a bridge in night.

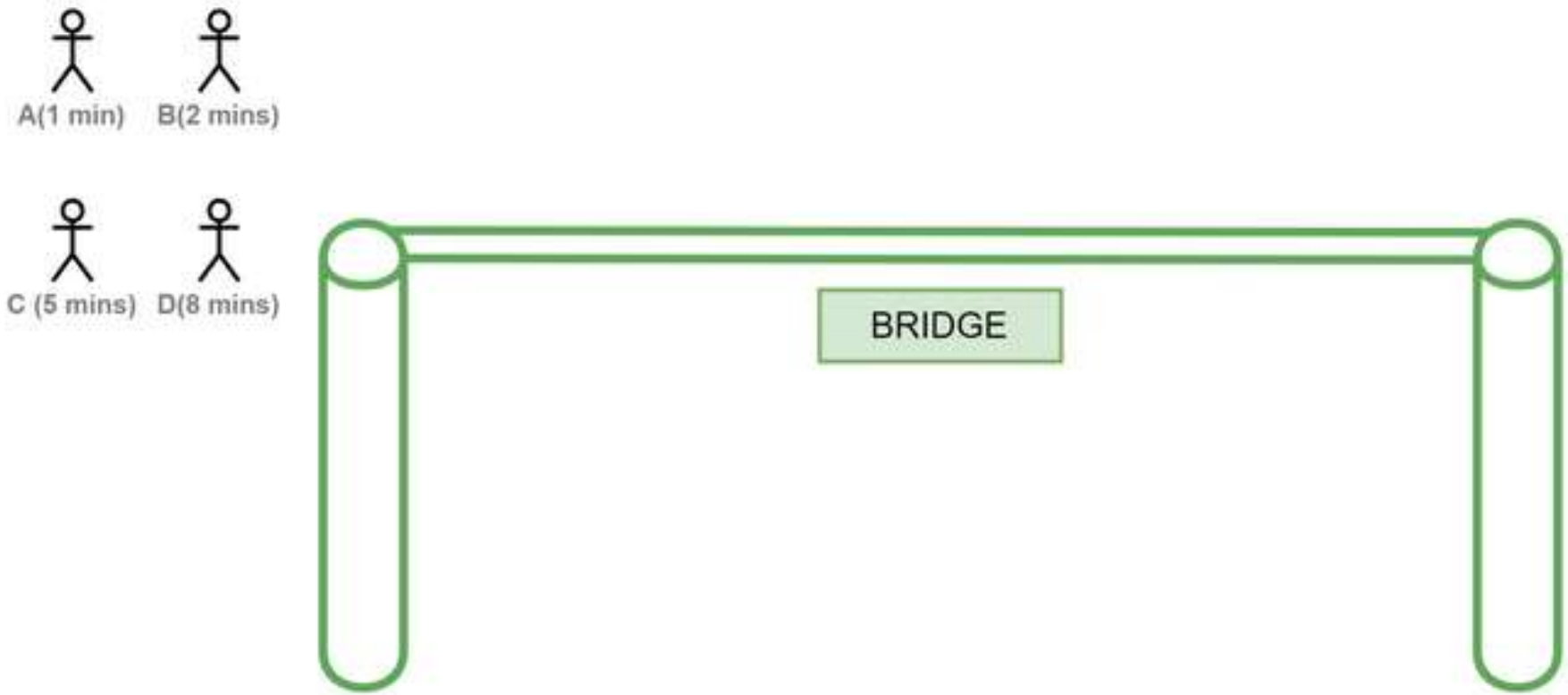
A takes 1 minute to cross the bridge.

B takes 2 minutes to cross the bridge.

C takes 5 minutes to cross the bridge.

D takes 10 minutes to cross the bridge.

There is only one torch with them and the bridge cannot be crossed without the torch. There cannot be more than two persons on the bridge at any time, and when two people cross the bridge together, they must move at the slower person’s pace.



A, B, C and D wants to cross the bridge. At most 2 people can cross the bridge at a single time

For example:

If person A and person D walk across first, 10 minutes have elapsed when they get to the other side of the bridge.

If person 4 returns the flashlight, a total of 20 minutes have passed

What is the minimum time possible to get them all across to the other side?

Puzzle time…

Torch and Bridge problem

There are 4 persons (A, B, C and D) who want to cross a bridge in night.

A takes 1 minute to cross the bridge.

B takes 2 minutes to cross the bridge.

C takes 5 minutes to cross the bridge.

D takes 8 minutes to cross the bridge.

There is only one torch with them and the bridge cannot be crossed without the torch. There cannot be more than two persons on the bridge at any time, and when two people cross the bridge together, they must move at the slower person’s pace.

Home work - 1

- Study on “Three methods for solving – computing **greatest common divisor** of two integers.