CS F211: DATA STRUCTURES & ALGORITHMS (2ND SEMESTER 2024-25) INTRODUCTION

Chittaranjan Hota, PhD
Sr. Professor of Computer Sc.
BITS-Pilani Hyderabad Campus
hota[AT]hyderabad.bits-pilani.ac.in

WHY SHOULD YOU STUDY THE COURSE?

You will be able to analyze a program to find out where to improve.

Searching your name in the MCN list.

Driving through Hyd using maps.

Image source: https://www.quora.com/

FIBONACCI: RECURSIVE OR ITERATIVE?

Algorithm 1: $F(n)$

Input: Some non-negative integer n
Output: The nth number in the Fibonacci Sequence
if $n \leq 1$ **then**
| **return** n
else
| **return** $F(n-1) + F(n-2)$;

Algorithm 2: $F(n)$

Input: Some non-negative integer n
Output: The nth number in the Fibonacci Sequence
$A[0] \leftarrow 0$;
$A[1] \leftarrow 1$;
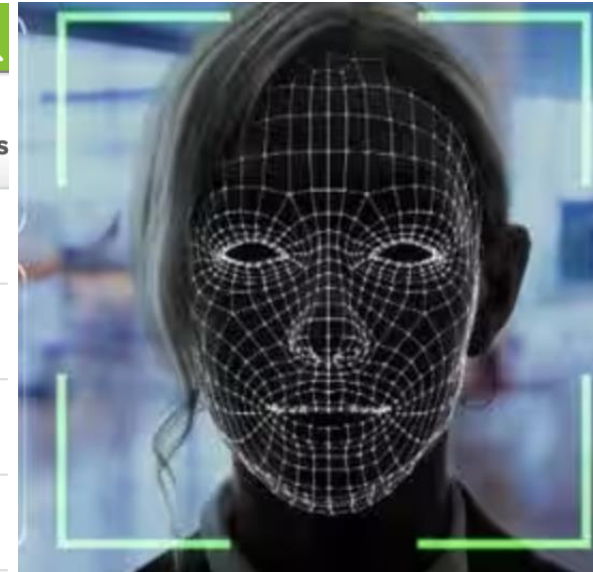**for** $i \leftarrow 2$ **to** $n-1$ **do**
| $A[i] \leftarrow A[i-1] + A[i-2]$;
return $A[n-1]$

Time Complexity: $O(2^n)$

Improved by memoization to: $O(n)$

Space complexity: $O(n)$

Time Complexity: $O(n)$

Space complexity: $O(1)$

WHAT KIND OF PROBLEMS CAN YOU SOLVE?
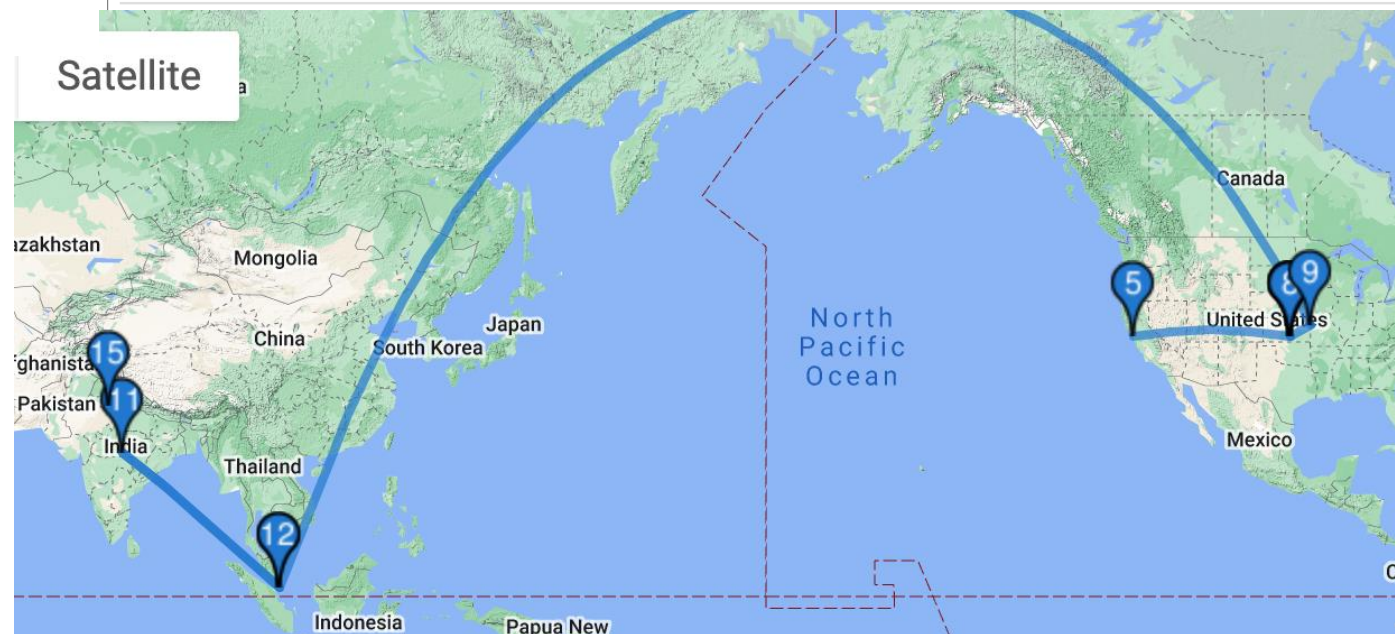
THE PROBLEM SOLVING PROCESS

Mathematical Model	**Abstract Data Types**	**Data Structures**
Informal Algorithm	**Pseudo-language Program**	**C++ Program**

If I had 1 hour to save the world, …

interface

implementation operations
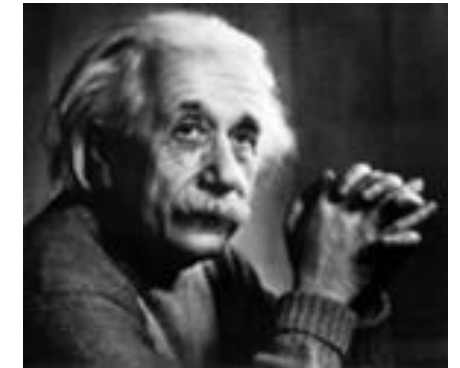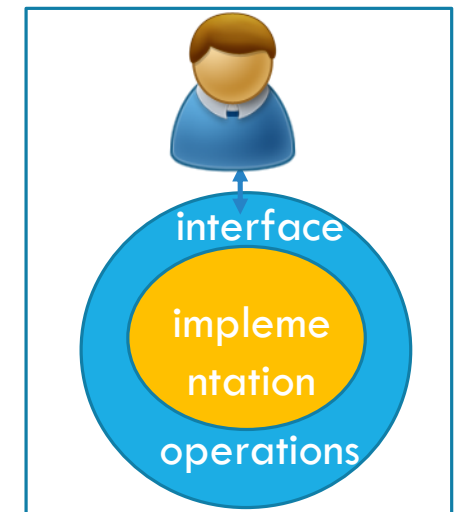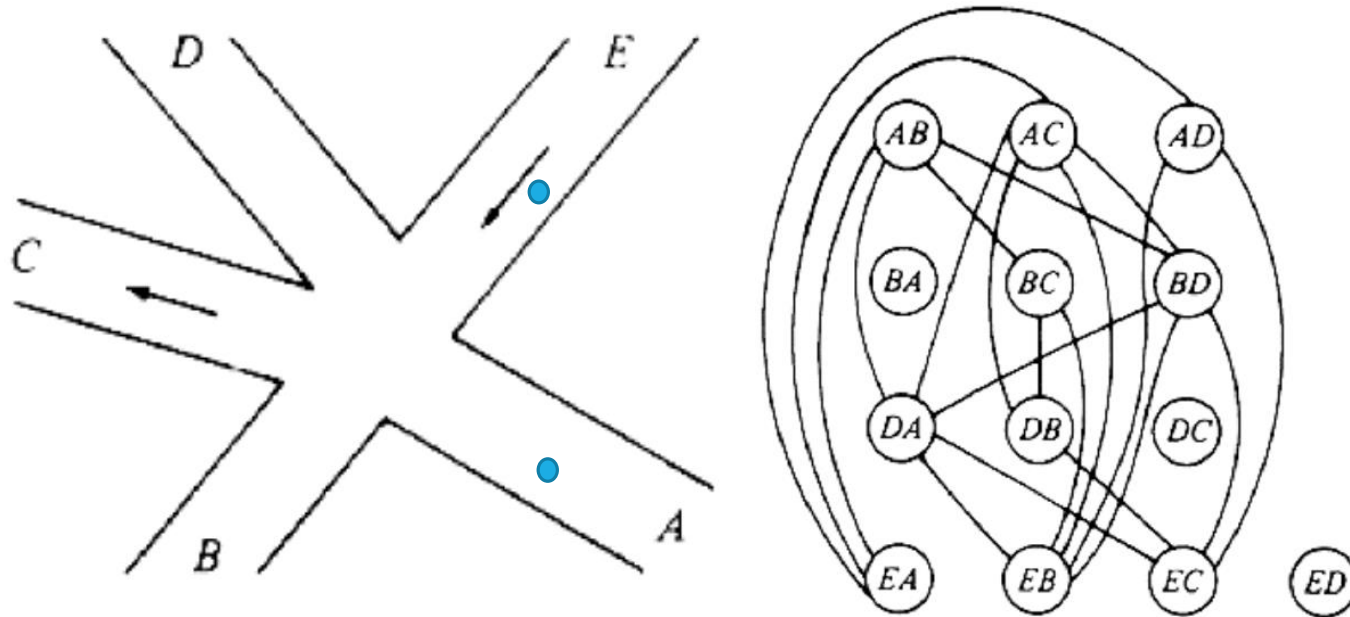
- finding currents in electrical circuits, predicting spread of covid-19

- Language translator

- An abstract data type (ADT), is a logical description of how we view the data and the operations that are allowed without knowing how they will be implemented.

- Data structure (physical description) is the implementation of ADTs.

AN EXAMPLE: A TRAFFIC LIGHTING SYSTEM

1. Select an uncolored vertex and color it with a new color.
2. Scan the list of uncolored vertices. For each uncolored vertex, determine whether it has an edge to any vertex already colored with the new color. If there is no such edge, color the present vertex with the new color.

(Problem of road intersection) (Graph with incompatible turns) (Greedy coloring algorithm)

- The approach is called "greedy" because it colors a vertex whenever it can, without considering the potential drawbacks inherent in making such a move. AB,AC,AD,BA,DC,ED; BC,BD,EA; DA,DB; EB,EC.

CONTINUED…

SET greedy_graph_coloring (Input:G:GRAPH,
Output: Newclr: SET) {
    Newclr ← Ø;
    for (each uncolored vertex 'v' ∈ G)
    {
        if 'v' is not adjacent to any vertex in Newclr
        {
            v ← colored;
            Newclr ← Newclr ∪ 'v'
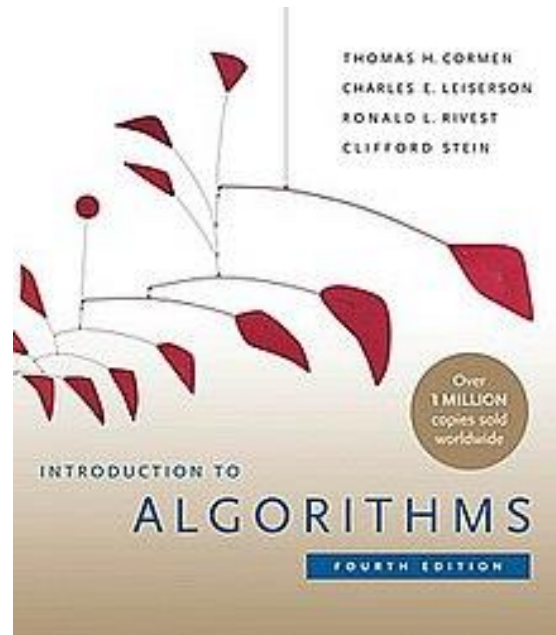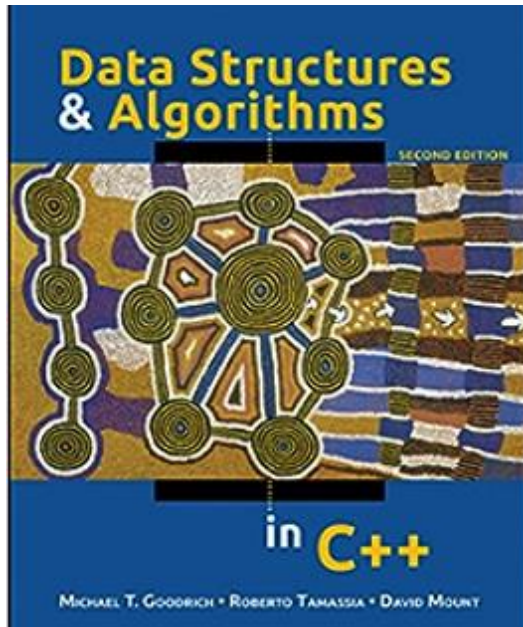   }
    }
}

(Pseudo Code)

GRAPH ADT: G
1. graphNew(): creating a graph
2. addVertex (v)
3. addEdge (v1, v2)
4. getVertex (unclored)
5. markVertex (colored)
6. …

	AB	AC	AD	BA	BC	BD	DA	DB	DC	EA	EB	EC	ED
AB					1	1	1			1			
AC						1	1	1		1	1		
AD										1	1	1	
BA													
BC	1							1			1		
BD	1	1					1				1	1	
DA	1	1				1					1	1	
DB		1			1								1
DC													
EA	1	1	1										
EB		1	1			1		1	1				
EC			1				1	1	1				
ED													

(Adjacency matrix to implement the graph)

COURSE CONTENT AND ADMINISTRATION

Text and Refs:

Course Content:

Intro to C++, Elementary data structures, Algorithm analysis techniques, More common data structures, Advanced data structures, Understanding algorithmic techniques.

Course notices and material: google class page   Chamber consultation hour: Every Monday (5 to 6 pm)

EVALUATIONS

Component	Duration	Weightage(%)	Date & Time	Nature of the component
Mid sem Test	1.5 hrs.	25%	03/03/2025 (2:00-3:30pm)	Closed Book
Lab Test (One)	1 hr.	10%	4th week of March, 2025 (in lab hours)	Open Book (Labsheets)
Lab quizzes (in every lab)	10 mins (best 10)	10%	-	Open Book (Labsheets)
Class quizzes (in theory class, weekly once)	10 mins (best 10)	15%	-	Open Book (Class notes)
Tutorial quizzes (in every tutorial class)	10 mins (best 10)	10%	-	Open Book (Tutorial notes)
Comprehensive examination	3 hrs.	30%	02/05/2025 (FN)	Part Open(20% CB +10%OB)

THANK YOU!

Next Class: Introduction to C++