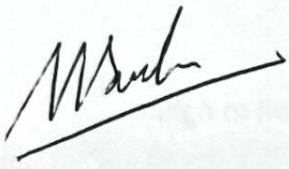



Final Examination

Date: 09/06/2023; Duration: 120 minutes

Open-book, only handwritten notes are allowed

SUBJECT: Algorithms & Data Structures (IT013IU)	
Approval by The SCSE Signature  Full name: Assoc. Prof. Nguyen Van Sinh	Lecturer: Signature  Full name: Trần Thanh Tùng
Proctor 1 Signature Full name:	Proctor 2 Signature Full name:
STUDENT INFO	
Student name: Student ID:	

INSTRUCTIONS: the total point is 100 (equivalent to 40% of the course)

1. *Purpose:*

- Test your knowledge on data structures and algorithms in the following topics: Binary Tree, Hash Table, Graphs, Advanced graph algorithms
- Examine your skill in analysis and design algorithms

2. *Requirement:*

- Write the answers and draw models CLEAN and TIDY

Note: For all calculations in this subject, the following **rounding convention** is used: $7/2 = 4$

1. Binary search tree (25pts)

Given a list of items, take items one by one from left to right.

Table 1 - Items

0 ✓	1 ✓	2 ✓	3 ✓	4 ✓	5 ✓	6 ✓	7 ✓	8 ✓	9 ✓	10 ✓	11 ✓	12 ✓	13 ✓	14 ✓	15 ✓
40	14	24	25	15	38	12	53	71	42	50	80	54	30	76	90

1.a. Insert all items into a binary search tree and draw the tree (15pts)

1.b. Delete the root node and redraw the tree (10pts)

2. Hash table (20pts)

Given a list of items in table 1 by taking items in the list from left to right

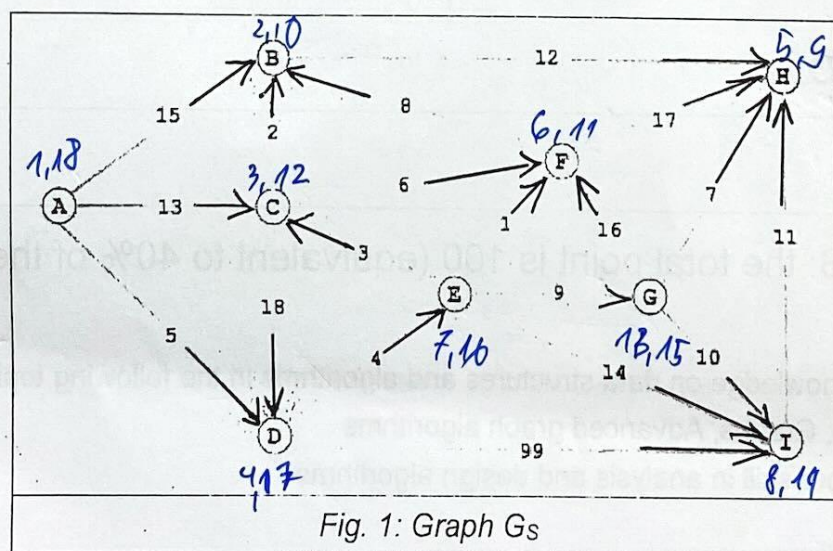
2.a. Insert all items into the hash table of size **29** by using the linear probing algorithm to solve collisions (10pts).

2.b. Change the hash table's size to **43**, redraw it (10pts)

3. Graph - Elementary Algorithms (30pts)

3.a Given the graph G_s in Fig. 1, run the BFS algorithm from **A** and redraw the graph with the values from the algorithm(15pts)

3.b. Find the topological sort order of all nodes in the graph G_s . Redraw the graph with the values from the algorithm(15pts)



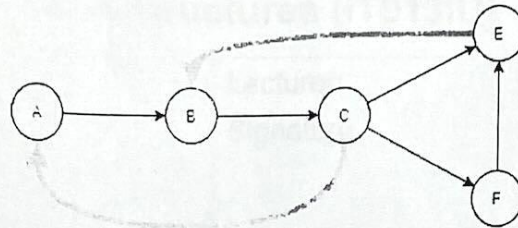
4. Algorithm – Binary Search Tree (15pts)

Implement an algorithm to sum all odd values in a Binary Search Tree

5. Algorithm to find cycle (10pts)

While traversing a graph using the DFS algorithm, backward edges are edges that link a node to another node in the path from the source node to the node. The backward edges form cycles.

For example, in the graph below, colored edges are backward edges. A – B – C, and B – C – E are simple cycles.



- (10pts) Propose an algorithm (write a pseudo-code) based on the DFS algorithm to print out all simple cycles of a given graph starting from a source node.

--- The end ---