United International University

Department of Computer Science and Engineering

Final Exam, Spring2024

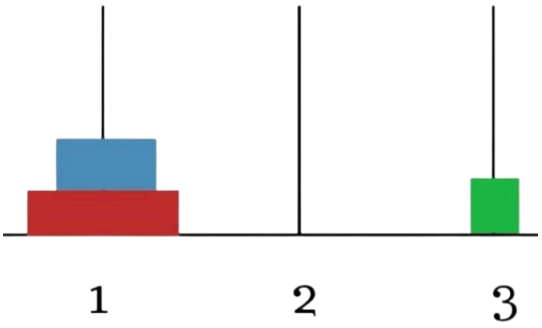
CSE 2215: Data Structure and Algorithms I

Total Marks: 40 Duration: 2 Hours

Any examinee found adopting unfair means will be expelled from the trimester/program as per UIU disciplinary rules.

Answer all the questions.

1. (a) What is the minimum number of moves does the below "Tower of Hanoi" problem need to transfer all the plates to the third stick? List out the sequence of disc transfers from one peg to another. [2]



(b) Consider a recursive function **int Hanoi(n, s, d, a).** [2]
Here:
n - represents the number of discs on the source peg.
s - denotes the source peg number.
d - denotes the destination peg number.
a - denotes the auxiliary peg number.
What will be the parameters and function calls if we want to solve the above (Question 1(a)) "Tower of Hanoi" problem using the function **Hanoi(n, s, d, a) only**?

(c) Find the values of A, B, C, and D for the **postfix expression AB+CD-/** to result in **3** after evaluation. Then, demonstrate the evaluation process **using a stack**. [4]

2. (a) In-order, Pre-order, and Post-order traversal algorithms are utilized to represent the structure and values of a tree visually. Consider a tree with seven nodes labeled as A, B, C, D, E, F, and G. Construct the tree such that: [2+2=4]
   I. The tree's Pre-order and In-order traversal sequences are identical.
   II. The tree's Post-order and In-order traversal sequences are identical.

(b) Assume that there is a binary search tree with no nodes. Perform the following operations on the tree: [4*2=8]
   I. Insert the values 100, 80, 81, 85, 400, 500, 201, 350, 360 and draw the tree
   II. Delete the node with the value 100 and draw the tree
   III. Determine the Successor of the node 201 from the tree (based on the last drawn diagram)
   IV. Write an algorithm to display all the internal nodes of a BST.

3. (a) Considering a binary heap containing n nodes, answer the following in terms of n: [0.5*4=2]

I. What is the time complexity of heapify()?

II. Write down the index of the parent of node i in a heap.

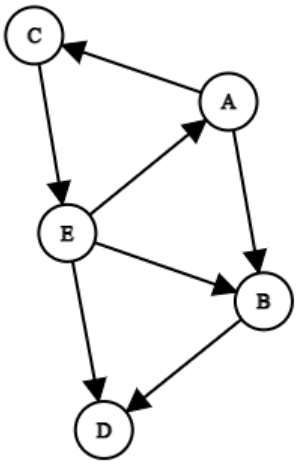
III. How many internal nodes does a heap have?

IV. Write down the index of the minimum valued node in a minheap.

(b) Show the steps of heapsort in descending order for the following numbers. The given heap is a Minheap. Just show the steps of sorting. Draw separate trees for each step and write down the final sorted sequence. [3]

5, 11, 8, 15, 19, 14, 27, 17

(c) Considering the following directed graph, answer the following questions. [1*3=3]

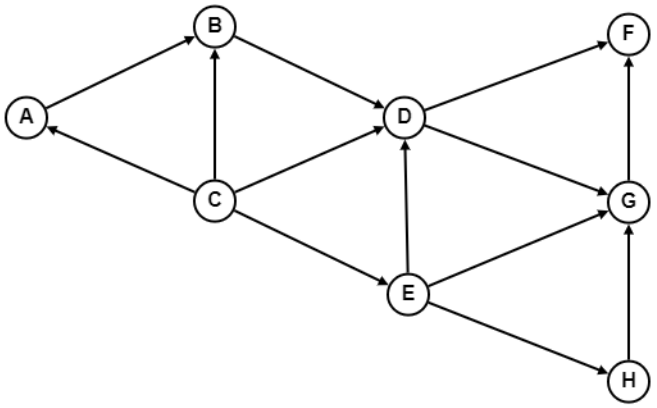


I. Draw a spanning tree from the given graph.

II. Using the graph as an example, show that $\sum_v indegree(v) = \sum_v outdegree(v) = m$, where m=total number of edges.

III. Represent the graph using an adjacency matrix.

4. (a) Apply Depth-First Search (DFS) algorithm on the following directed acyclic graph, and hence find out a topological ordering of the graph. [6]



(b) Look at the given pseudocode for the Breadth-First Search (BFS) algorithm, and answer the following questions. Here, there can be three states of a vertex - WHITE means the vertex is yet to be explored or discovered, GRAY means the vertex is discovered but not fully explored, and BLACK means the vertex is fully explored. [3+3=6]

```
BFS(G, s)
 1  for each vertex u ∈ V[G] − {s}
 2       do color[u] ← WHITE
 3          d[u] ← ∞
 4          π[u] ← NIL
 5  color[s] ← GRAY
 6  d[s] ← 0
 7  π[s] ← NIL
 8  Q ← ∅
 9  ENQUEUE(Q, s)
10  while Q ≠ ∅
11       do u ← DEQUEUE(Q)
12          for each v ∈ Adj[u]
13               do if color[v] = WHITE
14                     then color[v] ← GRAY
15                          d[v] ← d[u] + 1
16                          π[v] ← u
17                          ENQUEUE(Q, v)
18          color[u] ← BLACK
```

I. Suppose you are given a directed graph, and you have to apply BFS on it to find out if any cycle exists in the graph. Modify the pseudocode for this task. You only need to show the modification(s), and in/after which line(s) the modification(s) will occur.

II. What will be the time complexity of line 12 for a single vertex u, if you represent the graph as an adjacency matrix? What will be the time complexity if you use an adjacency list instead? Explain both answers briefly.