United International University (UIU)

Dept. of Computer Science and Engineering (CSE)

Final Examination Trimester: Summer 2023

Course: CSE 2215 Data Structure and Algorithms-I

Total Marks: 40, Time: 2 hours

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

There are FOUR questions. Answer all of them. Figures in the right-hand margin indicate full marks.

---

1. (a) Consider a recursive function int `Hanoi(n,s,d,a)`. [3]

Here:

`n` - represents the number of discs on the source peg.

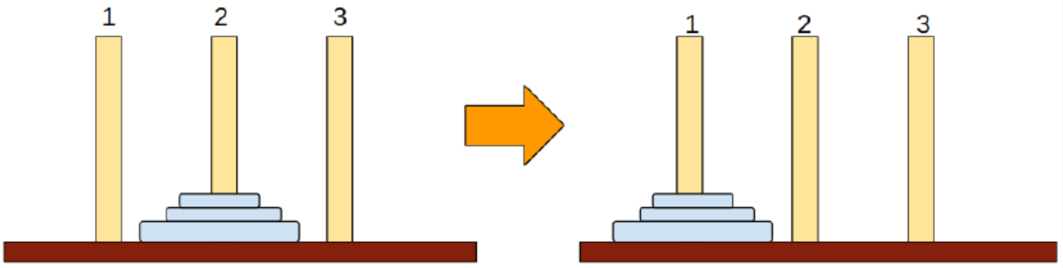
`s` - denotes the source peg number.

`d` - denotes the destination peg number.

`a` - denotes the auxiliary peg number.

The function returns the number of disc transfers required to move the discs from the source peg to the destination peg. It also prints the sequence of disc transfers.

Consider the following scenario: How would you determine the parameters `(n,s,d,a)` for the first call of the function `Hanoi`? Draw the corresponding recursive tree and list out the sequence of disc transfers from one peg to another.



(b) Your new trimester is beginning and you need to register for your new set of courses. While registering, you find that some courses have prerequisites. That is, some courses must be completed before others. The following is a list of prerequisites: [4]

SPL must be completed before DSA-1.
SPL must be completed before DSA-2.
DSA-1 must be completed before DSA-2.
DSA-1 must be completed before AI.
DSA-2 must be completed before AI.
DSA-2 must be completed before Graph Theory.

Which data structure will you use to represent this set of dependencies? Draw the representation. Find a valid order that will allow you to take the courses without violating any prerequisites.

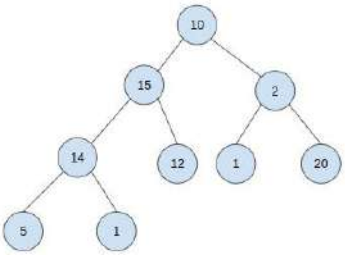
(c) Suggest a data structure for the flight landing system at Hazrat Shahjalal International Airport and justify your choice of data structure. [3]

Scenario 1: Flight BS01, BG02, IGO4, KAL696 and CS5 are lined up for landing in the order that is given.

Scenario 2: Flight BS01, BG02, IGO4, and CS5 are lined up for landing. Meanwhile, ATC (Air Traffic Controller) heard a Mayday (dual engine failure) call from Flight KAL696 an inbound flight from Kathmandu to Seoul which is traveling over Dhaka.

2. (a) Consider the given pseudocode of `Max-Heapify` and the following heap [4]

```
Max-Heapify(A,i)
    l = LEFT(i)
    r = RIGHT(i)
    if l<=A.heap-size and A[l]>A[i]
        largest=l
    else largest = i
    if r<=A.heap-size and A[r]>A[largest]
        largest = r
    if largest != i
        Exchange A[i] with A [largest]
        Max-Heapify(A,largest)
```



Does running `max-heapify` on the first node of the heap convert it into a **max heap**? Give a reason in favor of your answer.
If not, propose a suitable algorithm (along with the necessary pseudocode) to convert the given heap into a max heap.

(b) Draw a binary tree from the following **Preorder** and **Inorder** sequences [3]
Preorder: a b c d e f g h
Inorder: c b d a f g e h
Here, a=last two digits of your student id+3, b=a+3, e=x+b, c=b+e, d=a+2, f=c+d, h=111, g=121

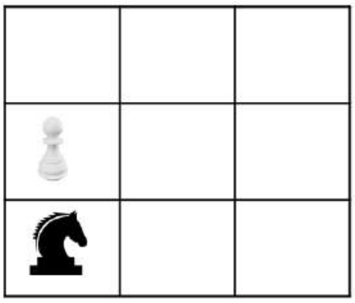
(c) Two disjoint sets `{y,p,z,x}` and `{r,t}` are given, where minimum one of a set is the representative of that set. Determine `UNION(Find(x), Find(t))`. How can you check x and y are in the same set using the Find operation? Here, x=last two digits of your student id+3, y=x+3, z=x+y, p=y+z, r=x+2, t=700. [3]

3. (a) Construct a binary search tree (**BST**) using the following nodes: [4]
30, 55, 10, 25, 45, 5, 20, 90, 70, 60, 75, 15, 20, 85, 80, 55.

(b) Traverse the constructed BST using the **inorder** traversal technique. [2]

(c) Delete 30 and 90 respectively from the BST. [4]

4.



(a) A knight moves on a chess board in an interesting way: it can jump two squares in one direction and then one square sideways. Now, draw a graph for the above chessboard in such a way that a vertex depicts each of the positions of the board. If the knight can move from one position on the board to another in a single move, draw an edge connecting the two corresponding vertices for those two positions. [4]

(b) What is the minimum number of moves required by the knight to capture the pawn in the above figure? Illustrate by simulating **BFS** on the graph obtained in **Question 4(a)**. [3]

(c) “If the pawn were at the center position in the middle of the board, the knight would never be able to capture the pawn” - Do you agree with this statement? Justify your answer using **BFS**. [3]