

Fall 2025: CS 3323A Final Exam

Total Time: 45 minutes

Total Points: 40

Write your name clearly. Answer all the questions.

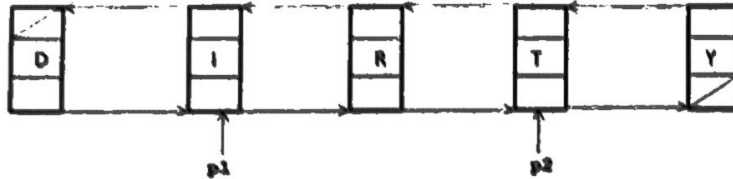
Name: _____ ID: _____ Date: _____

Question 1. True/False

[8]

1. Adjacency list performs better than Adjacency matrix to answer queries for direct connection between nodes.
2. Number of comparisons is equal to the length of the word in search operation in a Trie.
3. Performance of search in an ordered Tree depends on the shape of the Tree.
4. The value of null pointer is zero.
5. Insertion in Red-Black Tree will give black height imbalance problem but not double red.

Question 2. Assume the following doubly linked list with the two pointers p1 and p2: [8]



Find the value of each expression.

- (i) $P1 \rightarrow next \rightarrow data$
- (ii) $P1 \rightarrow next \rightarrow next$
- (iii) $P1 \rightarrow prev \rightarrow next$
- (iv) $P2 \rightarrow prev \rightarrow prev \rightarrow next \rightarrow data$
- (v) $P2 \rightarrow prev \rightarrow prev \rightarrow prev \rightarrow prev$

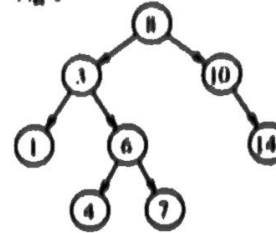
Question 3. Assume the structure of a Linked List node is as follows, write a recursive function to print the list in reverse manner. [8]

```

class Node
{ int data;
  Node *next; };
    
```

Question 4. Draw the resulting AVL tree, in each sequence of following operations such that resulting tree follows AVL property: [8]

Fig 1



After:

(i). Deleting 6

(ii) Deleting 8 (after i)

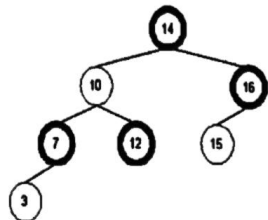
(iii) Inserting 8 (after i and ii)

(iv) Inserting 6 (after i, ii and iii)

Question 5. Consider an implementation of a hash table with five slots, with $h(x) = x \bmod 5$, and with collisions resolved by chaining. Show a final state of hash table happens when inserting the following sequence of keys into it: 35, 2, 18, 6, 3, 10, 8, 5 [4]

2. Delete in the original tree as shown in the figure
 - a. 15

Question 6. Observe following red black tree and show the tree after each following operations [6]



1. Insert in the original tree as shown in the figure

l. 2

3. Convert the original red black tree as shown in the figure into a 2-3-4 tree

Question 7. Provide Node structure of 2-4 Tree in C++ class format. [2]

Question 8 . Provide recursive C++ methods to compute the following in a BST that stores integer values. Prototype of the method is given as below: [5]

Search an item // returns true if the item exists other returns a false
 bool SearchAux (BinNode *node, int item)

Note: Above method will be called in Search as below:
 Bool Search(int item){ return SearchAux(myRoot, item);}