MCA-201: Data Structures and File Processing

Master of Computer and Applications

Semester II, May-2018

Time: Three Hours **Max. Marks: 70**

1. Convert the following infix expression to postfix expression

   a + (b + c * d + e) * f / g * (h + i * j – k) + m

   You are required to show the stack usage at each step. [10]

2. Give a node structure for class StringDoublyLinkedList, Write methods to do the following for the doubly linked list:

   - Insert a node at the beginning
   - Delete a node from the end
   - Traverse and print the values [10]

3. Draw the Binary Search Tree (BST) following each insertion from the following data values in the order given:

   42, 17, 89, 53, 72, 91, 3, 88, 90

   From the obtained BST delete 89 and then delete 17 (Draw the tree following each deletion). Explain all the deletion cases possible in a Binary Search Tree. [10]

4. Build an AVL tree from the following data. Show the tree every time you need to carry out a rotation (use predecessor for deletion choice)

   16, 21, 25, 9, 12, 6, 30, 38, 27

   From the final AVL tree obtained, remove node 25 and then remove 21. [10]

5. Build a min-heap for the following sequence of data in an array

   150 80 40 30 10 70 110 100 20 90 60 50 120 140 130

   Show the tree every time you need to adjust the keys in the structure being built. Show how the heap would appear after first three deletions. [10]

6. (a) Free memory blocks of size 60K, 25K, 12K, 20K, 35K, 45K and 40K are available in this order. Show the memory allocation for a sequence of job requests of size 22K,

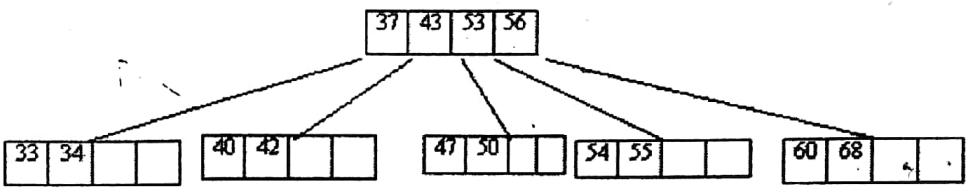
10K, 42K and 31K (in this order) in First Fit, Best Fit and Worst Fit allocation strategies. [5]

(b) Consider double hashing scenario where the hash table size, m is 10. The keys, k to be inserted are in the order given:

23, 91, 52, 40, 50, 60, 62

Hash functions are $h_1(k) = k \bmod m$ and $h_2(k) = 1 + (k \bmod (m-2))$
Double Hash Function, $DH(k, i) = (h_1(k) + i * h_2(k)) \bmod m$, where i= 0, 1, 2,.., m-1
Perform the insertion into the hash table. How many collisions occurred? [5]

7. (a) Consider the following B-tree of order 5

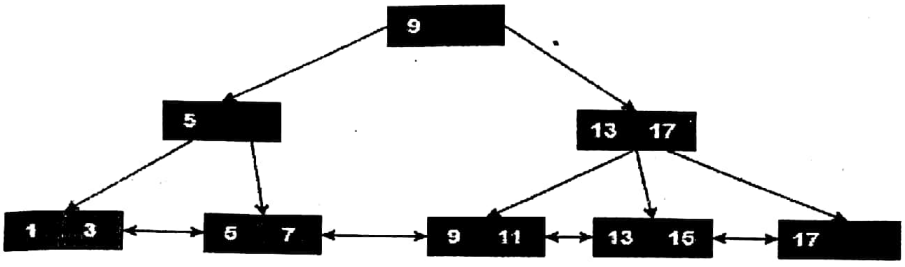


- Show the new tree after the insertion of the records with the key values 135, 200, 100 to the above tree if it's a proper tree structure.
- What is the min and max no of keys in B-Tree of order m?
- What is the min and max no of child/block pointer in B-tree of order m? [6]

(b) Consider a B-Tree with order P (order of a B-tree is defined as the max number of keys in a node). The block size is 1024 bytes and the size of search key is 6 bytes. The data/record pointer is 9 bytes and the block/child pointer is of 7 bytes. Find the max value for the order P of the B-Tree? [4]

or

(a) With reference to the B+ tree index of order 3 shown below, What is the minimum number of nodes (including the root node) that must be fetched in order to satisfy the following query: "Get all records with a search key greater than or equal to 7 and less than 15" ? Explain the answer. [3]



(b) Consider a B+ Tree with order P (order of a B+ tree is defined as the max number of keys in a node). The block size is 1024 bytes and the size of search key is 6 bytes. The data/record pointer is 9 bytes and the block/child pointer is of 7 bytes. Find the max value for the order P of the B+ tree for the internal node and leaf node? [7]