Computer Engineering Department, SVNIT, Surat.
End-Semester Examinations, November / December 2016
B.Tech-II (CO) –Semester III
Course: CO203 - Data Structures and Algorithms

Dated: 28th Nov, 2016 Time: 15:30 to 18:30 Max Marks: 100

Instructions:
1. Write your B.Tech Admission No/Roll No and other details clearly on the answer books while write your B.Tech Admission No on the question paper, too.
2. Assume any necessary data but give proper justifications.
3. Be precise and clear in answering the questions.

Q1 Answer the following. [Any Five] **[40]**

1) Suppose that a small company has given your group the contract of building a request-response application server. This means that the server should take an arriving request as input, process the request, and return the response as output. Some typical examples of such applications are searching on Google, accessing any database and railway booking counter. Assume that your team has 3 people and the work has been divided as follows: You are supposed to design the data structure, its interfaces and implementation, for handling the incoming requests. Another person will take the request(s) from your data structure, do the necessary processing and create the output. The third person will return the output as being the response corresponding the given request.
   a) Draw a picture of your data structure, and illustrate its use with an example. This will help you to figure out its working.
   b) Write the structure and function definitions (algorithm) to perform given task module.
2) Imagine that N people have decided to elect a leader by arranging themselves in a circle and eliminating every M^{th} person around the circle, till only one person remains, who becomes the leader.
   Now suppose you want the person at the K^{th} position to become the leader.
   a) Identify the data structure and Design an algorithm that will output M, given N and K. Clearly state any assumptions that you make and justify them.
   b) Show the working of your algorithm when N=7 and K=3.
3) Devise an algorithm to find the middle node in a singly linear linked list, using only a single pass and show the execution of your algorithm on an example.
4) Explain how you can implement a queue using a circular singly linked list. Show how you can do enqueue and dequeue operations in O(1) time.
   Note: A circular singly linked list is a singly linked list where the tail element points back to the head of the linked list (instead of pointing to a null element). The linked list is accessed by an external pointer pointing to one of the elements.
5) a) Determine and explain the functionality of the following code fragment:

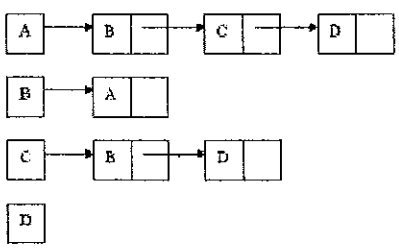
```
int function1 (int a[], int n, int x)
{
int i;
for(i = 0; i< n && a[i] != x; i++);
if(i == n) return -1;
else return i;
}
```

   b) Compute the best case, worst case and average case time complexity of the above function1. Explain your answers.
6) a) The integers given below are to be inserted in a hash table with 5 locations using chaining to resolve collisions. Construct hash table and use multiplicative hash method. 1,2,3,4,5,10,15,22,31,32,33,34,49,50,102.
   b) Show that, $\max\{f(n), g(n)\} = \theta\big(f(n) + g(n)\big)$.

Q2 Answer the following (Any Four): **[32]**

1. Show the AVL tree that results after each of the integer keys 9, 27, 50, 15, 2, 21, and 36 are inserted, in that order, into an initially empty AVL tree. Clearly show the tree that results after each insertion, and make clear any rotations that must be performed.
2. Consider the following alternative attempt at a recursive definition for a binary search tree with unique keys: a tree is a BST if (1) the key at the left child of the root is less than the key at the root which in turn is less than the key at the right child of the root and (2) the left subtree is a binary search tree and the right subtree is a binary search tree.
   Show that this definition is inadequate i.e. there are trees that satisfy this definition but are not binary search trees.
3. Write an algorithm shortest_path(G,s,d), for finding shortest path from the source vertex s to destination vertex d in a graph G.

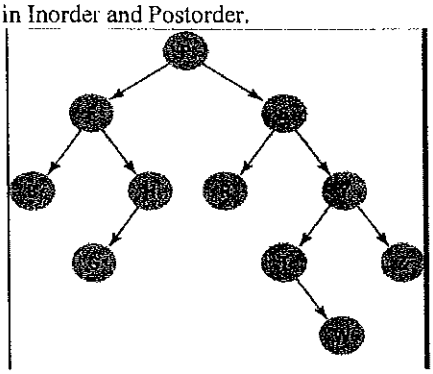
4. For each of the following applications, indicate which of data structures would be most suitable and give a brief justification for your choice. For data structures like trees and graphs, describe what information is stored in the vertices and edges, and, if the edges are weighted, describe what information is stored in the weights.
   a. Map of the Puget Sound highway system used to display traffic travel times on a web page. The map displays principle cities, intersections, and major landmarks, the roads that connect them, and the travel times between them along those roads. Travel times along the same road may be different in different directions.
   b. Chess board – an 8 x 8 board used for a game of chess. Each square on the board is either empty or contains a chess piece.
   c. A computer model showing the dependencies between the steps needed to assemble a B787 airplane at Boeing's Everett plant.
   d. The history list recording sites visited by the user of a web browser. As new sites are visited they are added to the list. The list also supports the operation of going back to the web page that was previously visited before the current page and going forward to the next page visited.
5. Following is an adjacency list representation of a directed graph where there are no weights assigned to the edges.



   a. Draw a picture of the directed graph that has the above adjacency list representation.
   b. Draw the adjacency matrix for the graph generated in (a).
   c. Let number of vertices in a graph be V=4 and each slot in the adjacency matrix requires 1 bit. What is the minimum amount of storage needed by an adjacency matrix representation of an undirected graph?
   d. For a graph with V vertices and E edges, how many items are required to be stored in adjacency list representation for.
      I. Directed graph
      II. Undirected graph

Q3 Answer the following (Any Seven): **[28]**

1. Index structures for large datasets cannot be stored in main memory. Storing it on a disk requires different approach to efficiency. Suggest a suitable data structure for this application with proper justification.
2. Derive the maximum number of items in a B-tree of order m and height h. What is the number of items when m = 5 and h = 2 ?
3. Discuss the applications of Breadth First and Depth First traversal of graph.
4. Traverse the following binary tree in Inorder and Postorder.



5. Given a binary tree whose root node address is given by a pointer variable T, write an algorithm that returns the address of the root node of a new tree which is the copy of the tree with root T.
6. Discuss pros and cons of the threaded storage representation for a binary search tree. Suggest a suitable node structure for the threaded storage representation for a binary tree.
7. Convert the following INFIX expressions into postfix and prefix:
   a. (4+8)*(6-5)/((3-2)*(2+2))
   b. (((1+2)*3)+6)/(2+3)
8. Draw an AVL tree of height 4 that contains the minimum possible number of nodes.