

NGM COLLEGE
DEPARTMENT OF COMPUTER SCIENCE(SF)
I-BSC COMPUTER SCIENCE(II SEMESTER)
18UCS205-DATA AND FILE STRUCTURE (2018-2021)
K1-LEVEL MULTIPLE CHOICE QUESTIONS

UNIT-I

1. What is the 'next' field of structure node in the Queue?

- a. Results into the storage of queue elements.**
- b. Results into the storage of address of next node by holding the next element of queue.**
- c. Results into the memory allocation of data elements to next node.**
- d. Results into the address allocation data elements to next node.**

ANS: B

2. From where does the insertion and deletion of elements get accomplished in Queues?

- a. Front & Rear ends respectively**
- b. Rear & Front ends respectively**
- c. Only Front ends**
- d. Only Rear ends**

ANS: B

3. Which among the below mentioned entities is / are essential for an Array Representation of a Queue?

- a. An array to hold queue elements**
- b. A variable to hold the index of front element**
- c. A variable to hold the index of rear element**
- d. All of the above**

ANS: D

4. What should be the value of rear (end) if the queue is full (elements are completely occupied)?

- a. 1**
- b. - 1**
- c. MAX + 1**
- d. MAX - 1**

e. zero (null)

ANS:D

5. Which among the below specified condition is applicable if the Queue is non - empty?

a. rear > front

b. rear < front

c. rear = front

d. Unpredictable

ANS:A

6. Where the elements are stored in accordance to the representation of Queue by a Linked Structure?

a. mesh

b. node

c. both a & b

d. none of the above

ANS:B

7. Which linear structure has a provision of Last-In-First-Out (LIFO) mechanism for its elements?

a. Stack

b. Queue

c. Both a & b

d. None of the above

ANS:A

8. Stacks do not find their applicability for _____.

a. Simplification of an arithmetic expression in postfix form

b. Recursion Implementation

c. Conversion of Infix to its equivalent Postfix Form

d. Allocation of Resources by an Operating System

ANS:D

9. Which data structures find their applications in BFS and DFS Traversal mechanisms on a Tree respectively?

a. Graph & Stack

b. Queue & Stack

c. Queue & Graph

d. None of the Above

ANS:B

10. How many elements are present in the stack if the variable top exhibits pointing towards the topmost element in the Array?

a. top +1

b. top - 1

- c. zero
- d. infinite

ANS:A

UNIT-II

11. Which expressions are also regarded as ' Reverse Polish Notations ' ?

- a. Prefix
- b. Postfix
- c. Infix
- d. All of the above

ANS:B

12. Which is the correct algorithmic sequence for the conversion of an expression from Infix to Prefix?

- A. Change of every '(' (opening bracket) by ')' (closing bracket) and vice-versa.
- B. Reversal of an infix expression.
- C. Conversion of the modified expression into postfix form.
- D. Reversal of postfix expression.

ANS:C

13. Which direction of scanning is suitable for the evaluation of a prefix expression?

- a. Left to Left
- b. Right to Right
- c. Left to Right
- d. Right to Left

ANS:D

14. When is the pop operation on Stack considered to be an error?

- a. Only when the stack is empty
- b. Only when the stack is full
- c. When the stack is neither empty nor full
- d. Cannot be predicted

ANS:A

15. What does the push operation $top = top + 1$ indicate while representing the stack in one - dimensional array?

- a. Stack Growing

- b. Stack Shrinking**
- c. Stack Stability**
- d. Stack Instability**

ANS:A

16. What happens if an expression tree reads the symbol in the form of an Operand?

- a. One node tree is created and a pointer is pushed towards it on the stack.**
- b. Pointer is pop to two trees in order to generate a new tree with root as its operator.**
- c. Both a & b**
- d. None of the Above**

ANS:A

17. Which type of linked list / s comprise / s a node containing a pointer to predecessor as well as successor?

- a. Doubly Linked List**
- b. Circular Linked List**
- c. Both a & b**
- d. None of the above**

ANS:C

18. Which balance factor is stored in the new field introduced by an AVL tree for the representation of a node?

- a. Length**
- b. Height**
- c. Width**
- d. Information**

ANS:B

19. Which type of linked list comprises the adjacently placed first and the last elements?

- a. Singly Linked List**
- b. Doubly Linked List**
- c. Circular Linked List**
- d. All of the above**

ANS:C

20. Which is the correct sequential order of constructing a binary tree for the expression $a + b * c + d * e$?

- a. A, B, C**
- b. B, C, A**
- c. B, A, C**
- d. C, A, B**

ANS:D

UNIT-III

21 How is an insertion of a node into an AVL tree carried out?

- a. By treating an AVL tree as a binary search tree**
- b. By updating the balance factors working upward from insertion point to the root**
- c. Both a & b**
- d. None of the Above**

ANS:C

22. What does a node possessing zero degree in Trees known as?

- a. Branch Node**
- b. Root Node**
- c. Leaf Node**
- d. Trunk Node**

ANS:C

23. Which of the following techniques represents the precise sequence of an In - Order Traversal of a Binary Tree?

- a. Visit the Root, Traverse Left Subtree, Traverse Right Subtree**
- b. Traverse Left Subtree, Visit the Root, Traverse Right Subtree**
- c. Traverse Left Subtree, Traverse Right Subtree, Visit the Root**
- d. None of the Above**

ANS:B

24. Traversal of a linked list always starts from the _____.

- a. First Node**
- b. Middle Node**
- c. Last Node**
- d. None of the Above**

ANS:A

25. Which balance factor is stored in the new field introduced by an AVL tree for the representation of a node?

- a. Length**
- b. Height**
- c. Width**
- d. Information**

ANS:B

26. What should we call a binary tree whose every node has either zero or two children?

- a. Complete Binary Tree**
- b. Binary Search Tree**
- c. Extended Binary Tree**
- d. None of Above**

ANS:C

27. Which of the following is the depth of a Complete Binary Tree?

- a. $D_n = n \log_2 n$**
- b. $D_n = n \log_2 n + 1$**
- c. $D_n = \log_2 n$**
- d. $D_n = \log_2 n + 1$**

ANS:D

28. If a binary tree of depth is “ d ”, then when can we say that it is an almost complete binary tree?

- a. Each leaf in the tree is either at level “ d ” or at level “ d – 1 ” .**
- b. For any node “ n ” in the tree with a right descendent at level “ d ” all the left descendants of “ n ” that are leaves, are also at level “ d ”.**
- c. Both A and B.**
- d. None of the above.**

ANS:C

29. If ' h ' is a hashing function and it is used to hash ' n ' keys into a table of size ' m ' where $n \leq m$. What is the expected number of collisions involving a particular key ' x ' ?

- a. less than 1.**
- b. less than n.**
- c. less than m.**
- d. less than $n / 2$.**

ANS:A

30. Which of the following is the faster operation in AVL Trees?

- a.** Insertion
- b.** Deletion
- c.** Updation
- d.** Retrieval

ANS:D

UNIT-IV

31. Key value pair is usually seen in _____.

- a.** Hash Tables
- b.** Heaps
- c.** Both a and b
- d.** Skip list

ANS:A

32. The operation of processing each element in the list is known as, _____.

- a.** Sorting
- b.** Merging
- c.** Inserting
- d.** Traversal

ANS:D

33. In a heap the element with the greatest key is always located in which node?

- a.** leaf
- b.** root
- c.** first node of left sub tree
- d.** first node of right sub tree

ANS:B

34. What should we call a binary tree whose every node has either zero or two children?

- a. Complete Binary Tree
- b. Binary Search Tree
- c. Extended Binary Tree
- d. None of Above

ANS:C

35. Which of the following is the fastest ways to store and retrieve data?

- a. Sorting
- b. Hashing
- c. Indexing
- d. both A and C

ANS:B

36. From where the insertion starts with a B tree?

- a. Root node
- b. Any node
- c. Could be A or B
- d. Leaf node

ANS:D

37. In a heap the element with the greatest key is always located in which node?

- a. leaf
- b. root
- c. first node of left sub tree
- d. first node of right sub tree

ANS:B

38. Which of the following is the faster operation in AVL Trees?

- a. Insertion
- b. Deletion
- c. Updation
- d. Retrieval

ANS:D

39.. _____ is a tree where each parent node have only one associated child node.

- a. Balanced Binary Tree
- b. Rooted Complete Binary Tree
- c. Complete Binary Tree
- d. Degenerate Tree

ANS:D

40.Which strategy of memory allocation yields the smallest free block size greater than or equal to the requested size?

- a. First - fit
- b. Best - fit
- c. Worst - fit
- d. Next - fit

ANS:B

UNIT-V

41. File organization that have no ordering of records, is called

- A. Heap file organization
- B. Clustered file organization
- C. Hashing file organization
- D. Sequential file organization

ANS:A

42.File organization in which records are stored in a logical order is known to be

- A. Clustered file organization
- B. Hashing file organization
- C. Sequential file organization
- D. Heap file organization

ANS:C

42. In hashing file organization, a hash function is calculated on some

- A. Attribute of each record
- B. Attribute of Method
- C. Attribute of relation
- D. Attribute of sequence

ANS:A

43.In sequential file organization, records are stored according to value of

- A. Record's entry
- B. Record's elimination
- C. Search key
- D. Function

ANS:C

43. Organization in which records are placed anywhere in file, where there is free space for record is referred to as

- A. Hashing file organization
- B. Sequential file organization
- C. Heap file organization
- D. Clustered file organization

ANS:C

44. A hash function must meet _____ criteria.

- a) Two
- b) Three
- c) Four
- d) None of the mentioned

ANS:B

45. In linear hashing, formula of file load factor is

- A. $l = rH (\text{bfr} * N)$**
- B. $l = rH (\text{bfr} + N)$**
- C. $l = rc (\text{bfr} - N)$**
- D. $l = rU (\text{bfr} * 2N)$**

ANS:A

46. Hashing technique which allocates fixed number of buckets is classified as

- A. dynamic hashing
- B. static hashing
- C. external hashing
- D. internal hashing

ANS:C

47. The file organization which allows us to read records that would satisfy the join condition by using one block read is

- a) Heap file organization**
- b) Sequential file organization**
- c) Clustering file organization**
- d) Hash file organization**

ANS:C

48. Index which has an entry for some of key value is classified as

- A. linear index
- B. dense index
- C. non dense index
- D. cluster index

ANS:C

49. Primary indexes, secondary indexes and cluster indexes are all types of

- A. ordered indexes

- B. unordered indexes
- C. linear indexes
- D. relative search indexes

ANS:A

50.Indexes which specifies address of records on disk with a physical pointer are classified as

- A. structural index
- B. hashing index
- C. physical index
- D. logical index

ANS:C

-

Department of Computer Science
18 UCS205 - Data and File Structure

K2 LEVEL QUESTIONS

Unit I

1. What is data structure?

Data structure refers to the way data is organized and manipulated. It seeks to find ways to make data access more efficient. When dealing with the data structure, we not only focus on one piece of data but the different set of data and how they can relate to one another in an organized manner.

2. Explain about file and structure storage structure.

The key difference between both the data structure is the memory area that is being accessed. When dealing with the structure that resides the main memory of the computer system, this is referred to as storage structure. When dealing with an auxiliary structure, we refer to it as file structures.

3

4. What is a linked list?

A linked list is a sequence of nodes in which each node is connected to the node following it. This forms a chain-like link for data storage.

5. State all the elements in a one-dimension array

To reference all the elements in a one -dimension array, you need to use an indexed loop, So that, the counter runs from 0 to the array size minus one. In this manner, You can reference all the elements in sequence by using the loop counter as the array subscript.

6. State the areas data structures are applied

Data structures are essential in almost every aspect where data is involved. In general, algorithms that involve efficient data structure is applied in the following areas: numerical analysis, operating system, A.I., compiler design, database management, graphics, and statistical analysis, to name a few.

7.What is LIFO?

LIFO is a short form of Last In First Out. It refers how data is accessed, stored and retrieved. Using this scheme, data that was stored last should be the one to be extracted first. This also means that in order to gain access to the first data, all the other data that was stored before this first data must first be retrieved and extracted.

8.What is a queue?

A queue is a data structure that can simulate a list or stream of data. In this structure, new elements are inserted at one end, and existing elements are removed from the other end.

9.What are binary trees?

A binary tree is one type of data structure that has two nodes, a left node, and a right node. In programming, binary trees are an extension of the linked list structures.

10.State data structures are applied when dealing with a recursive function

Recursion, is a function that calls itself based on a terminating condition, makes use of the stack. Using LIFO, a call to a recursive function saves the return address so that it knows how to return to the calling function after the call terminates.

Unit II

11.What is a stack?

A stack is a data structure in which only the top element can be accessed. As data is stored in the stack, each data is pushed downward, leaving the most recently added data on top.

12.Explain Binary Search Tree

A binary search tree stores data in such a way that they can be retrieved very efficiently. The left subtree contains nodes whose keys are less than the node's key

value, while the right subtree contains nodes whose keys are greater than or equal to the node's key value. Moreover, both subtrees are also binary search trees.

13. What are multidimensional arrays?

Multidimensional arrays make use of multiple indexes to store data. It is useful when storing data that cannot be represented using single dimensional indexing, such as data representation in a board game, tables with data stored in more than one column.

14. Are linked lists considered linear or non-linear data structures?

It depends on where you intend to apply linked lists. If you based it on storage, a linked list is considered non-linear. On the other hand, if you based it on access strategies, then a linked list is considered linear.

15. How does dynamic memory allocation help in managing data?

Apart from being able to store simple structured data types, dynamic memory allocation can combine separately allocated structured blocks to form composite structures that expand and contract as needed.

16. What is FIFO?

FIFO stands for First-in, First-out, and is used to represent how data is accessed in a queue. Data has been inserted into the queue list the longest is the one that is removed first.

17. What is an ordered list?

An ordered list is a list in which each node's position in the list is determined by the value of its key component, so that the key values form an increasing sequence, as the list is traversed.

18. What is merge sort?

Merge sort, is a divide-and-conquer approach for sorting the data. In a sequence of data, adjacent ones are merged and sorted to create bigger sorted lists. These sorted lists are then merged again to form an even bigger sorted list, which continues until you have one single sorted list.

19. State the difference between NULL and VOID

Null is a value, whereas Void is a data type identifier. A variable that is given a Null value indicates an empty value. The void is used to identify pointers as having no initial size.

20.State the primary advantage of a linked list?

A linked list is an ideal data structure because it can be modified easily. This means that editing a linked list works regardless of how many elements are in the list.

Unit III

21.State the difference between a PUSH and a POP?

Pushing and popping applies to the way data is stored and retrieved in a stack. A push denotes data being added to it, meaning data is being “pushed” into the stack. On the other hand, a pop denotes data retrieval, and in particular, refers to the topmost data being accessed.

22. Define linear search

A linear search refers to the way a target key is being searched in a sequential data structure. In this method, each element in the list is checked and compared against the target key. The process is repeated until found or if the end of the file has been reached.

23.How does a selection sort work for an array?

The selection sort is a fairly intuitive sorting algorithm, though not necessarily efficient. In this process, the smallest element is first located and switched with the element at subscript zero, thereby placing the smallest element in the first position.

24.How do signed and unsigned numbers affect memory?

In the case of signed numbers, the first bit is used to indicate whether positive or negative, which leaves you with one bit short. With unsigned numbers, you have all bits available for that number. The effect is best seen in the number range (an unsigned 8-bit number has a range 0-255, while the 8-bit signed number has a range -128 to +127).

25.What is the minimum number of nodes that a binary tree can have?

A binary tree can have a minimum of zero nodes, which occurs when the nodes have NULL values. Furthermore, a binary tree can also have 1 or 2 nodes.

26. What are dynamic data structures?

Dynamic data structures are structures that expand and contract as a program runs. It provides a flexible means of manipulating data because it can adjust according to the size of the data.

27. State the data structures where pointers are applied?

Pointers that are used in linked list have various applications in the data structure. Data structures that make use of this concept include the Stack, Queue, Linked List and Binary Tree.

28. What are ARRAYS?

When dealing with arrays, data is stored and retrieved using an index that refers to the element number in the data sequence. This means that data can be accessed in any order. In programming, an array is declared as a variable having a number of indexed elements.

29. What is the minimum number of queues needed when implementing a priority queue?

The minimum number of queues needed in this case is two. One queue is intended for sorting priorities while the other queue is used for actual storage of data.

30. Which sorting algorithm is considered the fastest?

There are many types of sorting algorithms: quick sort, bubble sort, balloon sort, radix sort, merge sort, etc. Not one can be considered the fastest because each algorithm is designed for a particular data structure and data set. It would depend on the data set that you would want to sort.

Unit IV

31. Differentiate STACK from ARRAY.

Stack follows a LIFO pattern. It means that data access follows a sequence wherein the last data to be stored when the first one to be extracted. Arrays, on the other hand, does not follow a particular order and instead can be accessed by referring to the indexed element within the array.

32. What is a dequeue?

A dequeue is a double-ended queue. This is a structure wherein elements can be inserted or removed from either end.

33. What are the parts of a linked list?

A linked list typically has two parts: the head and the tail. Between the head and tail lie the actual nodes. All these nodes are linked sequentially.

34. What is a graph?

A graph is one type of data structure that contains a set of ordered pairs. These ordered pairs are also referred to as edges or arcs and are used to connect nodes where data can be stored and retrieved.

35. Differentiate linear from a nonlinear data structure.

The linear data structure is a structure wherein data elements are adjacent to each other. Examples of linear data structure include arrays, linked lists, stacks, and queues. On the other hand, a non-linear data structure is a structure wherein each data element can connect to more than two adjacent data elements. Examples of nonlinear data structure include trees and graphs.

36. What is an AVL tree?

An AVL tree is a type of binary search tree that is always in a state of partially balanced. The balance is measured as a difference between the heights of the subtrees from the root. This self-balancing tree was known to be the first data structure to be designed as such.

37. What are doubly linked lists?

Doubly linked lists are a special type of linked list wherein traversal across the data elements can be done in both directions. This is made possible by having two links in every node, one that links to the next node and another one that connects to the previous node.

38. What is Huffman's algorithm?

Huffman's algorithm is used for creating extended binary trees that have minimum weighted path lengths from the given weights. It makes use of a table that contains the frequency of occurrence for each data element.

39. What is Fibonacci search?

Fibonacci search is a search algorithm that applies to a sorted array. It makes use of a divide-and-conquer approach that can significantly reduce the time needed in order to reach the target element.

40. Briefly explain recursive algorithm.

Recursive algorithm targets a problem by dividing it into smaller, manageable sub-problems. The output of one recursion after processing one sub-problem becomes the input to the next recursive process.

3. When is a binary search best applied?

A binary search is an algorithm that is best applied to search a list when the elements are already in order or sorted. The list is searched starting in the middle, such that if that middle value is not the target search key, it will check to see if it will continue the search on the lower half of the list or the higher half. The split and search will then continue in the same manner.

Unit V

41. How do you search for a target key in a linked list?

To find the target key in a linked list, you have to apply sequential search. Each node is traversed and compared with the target key, and if it is different, then it follows the link to the next node. This traversal continues until either the target key is found or if the last node is reached.

42. What is the difference between file structure and storage structure?

Difference between file structure and storage structure:

The main difference between file structure and storage structure is based on memory area that is being accessed.

Storage structure: It is the representation of the data structure in the computer memory.

File structure: It is the representation of the storage structure in the auxiliary memory.

43. List the data structures which are used in RDBMS, Network Data Model, and Hierarchical Data Model.

- RDBMS uses Array data structure
- Network data model uses Graph
- Hierarchical data model uses Trees

44. Which data structure is used to perform recursion?

Stack data structure is used in recursion due to its last in first out nature. Operating system maintains the stack in order to save the iteration variables at each function call

45. What are the operations that can be performed on a stack?

- Push Operations
- Pop Operations
- Peek Operations

46. Write the stack overflow condition.

Overflow occurs when **top = Maxsize - 1**

47. State the difference between PUSH and POP?

PUSH and POP operations specify how data is stored and retrieved in a stack.

PUSH: PUSH specifies that data is being "inserted" into the stack.

POP: POP specifies data retrieval. It means that data is being deleted from the stack.

48. What is a multidimensional array?

The multidimensional array can be defined as the array of arrays in which, the data is stored in tabular form consists of rows and columns. 2D arrays are created to implement a relational database lookalike data structure. It provides ease of holding the bulk of data at once which can be passed to any number of functions wherever required.

49. State the elements of a 2D array are stored in the memory?

There are two techniques by using which, the elements of a 2D array can be stored in the memory.

Row-Major Order

Column-Major Order

50. Define Linked List Data structure.

Linked List is the collection of randomly stored data objects called nodes. In Linked List, each node is linked to its adjacent node through a pointer. A node contains two fields, i.e. Data Field and Link Field.

NGM COLLEGE
DEPARTMENT OF COMPUTER SCIENCE(SF)
I-BSC COMPUTER SCIENCE(II SEMESTER)
18UCS205-DATA AND FILE STRUCTURE (2018-2021)
K3-LEVEL MULTIPLE CHOICE QUESTIONS

UNIT-I

1. Relate the need of algorithm analysis?
2. Compute the criteria of algorithm analysis?
3. Examine ordered list in data structure?
4. Experiment the representation of arrays.
5. Sketch the reference of all the elements in a one-dimension array?
6. Assess the fundamentals of stacks.
7. Predict the use queues?
8. Find the operations can be performed on Queues?
9. Show the evaluation of expression.
10. Solve multiple stacks and queues.

UNIT-II

1. Construct linear searching in binary tree.
2. Organise singly linked list in detail.
3. Use of linked stacks and queues.
4. Predict circular linked lists in data structure.
5. Show the functions of linked list.
6. Experiment following Infix expression into Postfix expression using Tabular method. $a - b / c * d + e * f / g$.
7. Compute doubly linked lists in data file structure.
8. Develop polynomial addition

9.. Organise Dynamic storage management.

10. Sketch garbage collection.

UNIT-III

1..Examine a binary search tree?

2.Illustrate tree traversal in data file structure.

3. Demonstrate AVL Tree with example.

4.Complete a skewed binary tree with height balancing factor.

5. Operate heap sorting in data structure?

6. Apply linear probing is hashing?

7.Show static tree table.

8.Relate dynamic tree table with rotations.

9.Develop the Hashing Functions based on the various methods by which the key value is found.

10. Predict the types of Collision Resolution Techniques and the methods used in each of the type?

UNIT-IV

1. Complete basic algorithm for searching a binary search tree.

2. Manipulate an algorithm for merge sort technique. Illustrate with an example. Give its complexity.

3. Illustrate the insertion sort algorithm and bubble sort algorithm on input [30,20,10,60,70,40]

4. Organise an algorithm to perform binary search. Using the algorithm, search for elements 23 and 47 in the given set of elements[12 23 27 35 39 42 50].

5. Apply Quick Sorting with the help of example

6. Construct Radix Sort with the help of example.

7. Interpret Searching? Explain Sequential search and Binary search with help of example.

8. Construct 2-way merge sort and how it works?
9. Predict linear searching with example.
10. Explain Fibonacci search with example.

UNIT-V

1. State different File Organizations and discuss the advantages and disadvantages of each of them.
2. prepare a note on indexed file organization.
3. Operate Sequential File Structure in detail.
4. Examine the four query types in files.
5. calculate various fundamental file organization techniques and explain each in brief.
6. Modify collision. What is linear probing? The following keys 10, 16, 11, 1, 3, 4, 23 and 15 are inserted into an initially empty hash table of length 10 using open addressing with hash function $h(k) = k \text{ mod } 10$ and linear probing. What is the resultant hash table?
7. Find hashing. What are the properties of a good hash function? With necessary examples explain four different hashing techniques.
8. Choose Direct access indexing with example.
9. Develop the basic two techniques for Collision-resolution in Hashing with example.
10. Practice various multiple key access file organization in brief with advantages and disadvantages of each method.

NGM COLLEGE
DEPARTMENT OF COMPUTER SCIENCE(SF)
I-BSC COMPUTER SCIENCE(II SEMESTER)
18UCS205-DATA AND FILE STRUCTURE (2018-2021)
K4&K5-LEVEL MULTIPLE CHOICE QUESTIONS

UNIT-I

1. Explain Stack and Queue data structure .
2. Formulate the difference between a Stack and an Array?
3. Distinguish creation and analysis of programs.
4. Outline an algorithm for evaluating a postfix expression and evaluate the following postfix expression using the algorithm $AB+CD/AD-EA^+*$ where $A=2, B=7, C=9, D=3, E=5$.
5. Develop multiple stacks and queues in arrays.

UNIT-II

1. How a stack can be implemented using linked list
2. Analyse circularly linked lists and singly linked list.
3. Identify polynomial addition using stack.
4. Determine allocation and freeing nodes.
5. Analyse compaction algorithm .

UNIT-III

1. Integrate following with reference to trees. (i) Height of the tree (ii) Complete Binary Tree (iii) Expression tree (iv) Sibling (v) Full Binary Tree
2. Examine height balanced tree? How re-balancing is done in height balanced tree.
3. Generate cost of a tree using Huffman algorithm.

4. Identify the structure of hash table.
5. Illustrate the functions of hash table.

UNIT-IV

1. Classify searching and sorting with examples.
2. Formulate the binary search techniques.
3. Distinguish Insertion and quick sorting.
4. Outline Heap sorting with tree structure.
5. Develop radix sorting with iterations.

UNIT-V

1. Relate queries and its types in files.
2. Explain sequential organizations in detail.
3. Revise cylinder-surface indexing.
4. Classify linked organization in detail.
5. Summarize storage management in data file structure.