**USN**

**21CS32**

**First/Third Semester B.E. / B.Arch. Semester End Examination, January-March2022-23**

**Data Structures and Algorithms**

**Time: 3 Hours Max. Marks: 100**

|  |  |  |
| --- | --- | --- |
| ***Instructions:*** | ***1.*** | ***From Part A answer any 5 questions each Question Carries 6 Marks.*** |
|  | ***2.*** | ***From Part B answer any one full question from each unit and each Question Carries 10 Marks.*** |
|  | ***3.*** | ***From Part C answer any one full question and each Question Carries 20 Marks.*** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PART A** | | | | | | |
| **Answer any Five.** | | | **L** | **CO** | **PO** | **M** |
| **1.** | Outline the algorithm to delete data from the queue and retrieve the data at the front of the queue | | **(L2)** | **1** | **2** | **(6)** |
| **2.** | Explain pointer to function concept with example | | **(L2)** | **1** | **2** | **(6)** |
| **3.** | Explain the dynamic memory management functions in C | | **(L2)** | **1** | **2** | **(6)** |
| **4.** | Outline the algorithm to destroy the queue | | **(L2)** | **1** | **2** | **(6)** |
| **5** | Compare and contrast linked list and array. | | **(L2)** |  |  | **(6)** |
| **6.** | What is binary Tree? Define the following terminologies of binary tree   1. Root b) leaf node c) internal node d) out degree e) in degree f) degree | | **(L2)** | **1** | **2** | **(6)** |
| **7.** | Explain the basic stack operations. | | **(L2)** | **1** | **2** | **(6)** |
| **PART B(minimum L3 level questions)** | | | | | | |
|  |  | **UNIT - I** | **L** | **CO** | **PO** | **M** |
| **8** | a. | Identify the data structure required to convert the given infix expression to postfix expression. And apply the same to convert the following expression.   1. A + B – ( C \* (D – E) / ( X \* Y) ) 2. ( 2\* 3 + 4 ^ ( 5 / 6) ) | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3)** | **(5)** |
|  | b. | Develop a C program to print the elements of array in forward and reverse direction using pointers | | | | |
|  |  |  | **(L3)** | **( 2 )** | **(3 )** | **( 5 )** |
|  |  | OR |  |  |  |  |
| **9** | a. | Develop a C program using stack to evaluate the given infix expression. | | | | |
|  |  |  | **(L3)** | **( 2)** | **( 3 )** | **( 5 )** |
|  | b. | Develop a program in C to display the result of n students by reading n students name, usn and marks out of 50 in three subjects from user. Students pass the exam only if the average marks is more than 40. Read number of students “n” from user. | | | | |
|  |  |  | **(L3)** | **(2 )** | **(3 )** | **(5 )** |
|  |  | **UNIT – II** | **L** | **CO** | **PO** | **M** |
| **10** | a. | Develop a C program to implement all basic operations of circular queue to store integer numbers | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3 )** | **(5 )** |
|  | b. | Develop a C program to traverse an ordered list implemented using a linked list and delete all thenodes whose keys are negative | | | | |
|  |  |  | **(L3)** | **( 2 )** | **(3 )** | **( 5 )** |
|  |  | OR |  |  |  |  |
| **11** | a. | Explain the algorithm to insert a node in the list and apply the same to perform an ordered insertion of a node with value 45 in the given list. | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3 )** | **( 5 )** |
|  | b. | Develop a C program to implement all basic operations of queue to store floating point numbers | | | | |
|  |  |  | **(L3)** | **( 2 )** | **(3 )** | **( 5 )** |
|  |  | **UNIT - III** | **L** | **CO** | **PO** | **M** |
| **12** | a. | Construct a binary search tree for the following number sequence and write a traversal function for obtaining the sorted number sequence from the constructed tree.  34,23,12,22,25,45,67,2,13,67,76,56,51,44,123,89,76,66,29 | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3 )** | **( 5 )** |
|  | b. | Given Inorder and Preorder traversals construct the binary tree  Inorder sequence: D B E A F C Preorder sequence: A B D E C F | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3 )** | **( 5 )** |
|  |  | OR |  |  |  |  |
| **13** | a. | Consider the following algorithm.  Algorithm Mystery(A[0..n − 1, 0..n − 1])  //Input: A matrix A[0..n − 1, 0..n − 1] of real numbers  for i← 0 to n − 2 do  for j ←i + 1 to n − 1 do  if A[i, j] ≠ A[j, i]  return false  return true  a. What does this algorithm compute?  b. What is its basic operation?  c. Set up sum expression for the algorithms basic operation and analyze the order of growth of an algorithm.  d. What is the efficiency class of this algorithm? | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3 )** | **( 5 )** |
|  | b. | Compare the order of growth of   1. log2n and √n 2. n! and 2n | | | | |
|  |  |  | **(L3)** | **( 2 )** | **( 3 )** | **( 5 )** |
|  |  | **UNIT - IV** | **L** | **CO** | **PO** | **M** |
| **14** | a. | Apply quick sort to sort “MERGESORT”. Draw the recursive tree. Discuss the order of growth of an algorithm. | | | | |
|  |  |  | **(L3)** | **( 3 )** | **(3 )** | **(5)** |
|  | b. | Solve the following recurrence relation and find the order of growth of an algorithm.  T(n)=4T(n/2)+n, T(1)=1 | | | | |
|  |  |  | **(L3)** | **(3 )** | **( 3 )** | **(5 )** |
|  |  | OR |  |  |  |  |
| **15** | a. | What is Spanning Tree? Apply Prim’s algorithm to the following graph and find minimum cost spanning tree.  C:\Users\HP\Downloads\1.JPG | | | | |
|  |  |  | **(L3)** | **(3 )** | **(3 )** | **(5)** |
|  | b. | Discuss different ways of representing a graph. Analyze and comment on which data structure is efficient to represent the dense graph and the sparse graph. | | | | |
|  |  |  | **(L3)** | **( 3 )** | **( 3 )** | **(5 )** |
|  |  | **UNIT -V** | **L** | **CO** | **PO** | **M** |
| **16** | a. | What is dynamic programming? Apply Floyd’s algorithm to find all pairs shortest path of the given graph.  Floyd - Warshall Algorithm | | | | |
|  |  |  | **(L3)** | **( 3 )** | **(3 )** | **( 5 )** |
|  | b. | Explain how Branch and Bound works. Apply Branch and Bound technique to find the solution for the job assignment problem. Draw state space tree.  Job Assignment Problem using Branch And Bound - GeeksforGeeks | | | | |
|  |  |  | **(L3)** | **( 4)** | **(3 )** | **( 5 )** |
|  |  | OR |  |  |  |  |
| **17** | a. | Apply Dynamic Programming to solve the following knapsack problem. Write the items included in the knapsack using backtracking method. Knapsack Capacity is W=16.     |  |  |  | | --- | --- | --- | | Item | Weight | Value | | 1 | 6 | 12 | | 2 | 3 | 27 | | 3 | 7 | 34 | | 4 | 8 | 12 | | 5 | 5 | 25 | | | | | |
|  |  |  | **(L3)** | **( 4 )** | **(3 )** | **( 5 )** |
|  | b. | Solve the following Subset sum problem applying Backtracking strategy. Write all possible solutions where S= { 2,5,3,7,10} and d=10. Create State Space Tree. | | | | |
|  |  |  | **(L3)** | **(4 )** | **(3 )** | **( 5 )** |
| **PART C** | | | **L** | **CO** | **PO** | **M** |
| **18** | a. | Analyze the code given below and make the necessary changes in the code to get the tree traversal output for the binary tree as shown in the sample output:  void print\_tree(struct node \*root)  {  struct node \* temp;  while(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_) //missing conditional statement  {  printf(“%c ”, temp);  if(temp->left!=null)  enqueue(temp->left);  if(temp->right!=null)  enqueue(temp->right);  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //missing statement  }  }  **Expected Output: K L S G I T** |  |  |  |  |
|  |  |  | **L4** | **2** | **3** | **20** |
|  |  | OR |  |  |  |  |
| **19** | a. | Given QSIZE = 3. Analyse the status of the queue after every operation given below.   1. display() 2. enqueue(5) 3. enqueue(8) 4. dequeue() 5. enqueue(7) 6. enqueue(9) 7. enqueue(1) 8. dequeue() 9. dequeue() 10. enqueue(3) |  |  |  |  |
|  |  |  | **(L4)** | **(2 )** | **(2 )** | **(10 )** |
|  | b. | Assume that the data of voters of a constituency is captured with the following fields (voterID, Name, Age, Gender, Caste, Subcaste, Address), Write a function to demonstrate the use of BST to find the count for each age value in the entire constituency.  **Expected Output:**   |  |  | | --- | --- | | Age | Count | | 18 | 1560 | | 19 | 2089 | | 20 | 5908 | | …. | …. | | 100 | 4 |   Write a function to use the data generated by the above function and display lowest and highest count of voters and their age.  **Expected Output:**  Highest Count: 5908 Age: 20  Lowest Count: 4 Age: 100 |  |  |  |  |
|  |  |  | **(L4)** | **( 2 )** | **( 3 )** | **(10)** |