|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | **Reg. No.:** | |  | | | |  |
| **Name :** | |  | | | |  |
|  | | | | | | | | | |  |
| **Mid-Term Examinations, August 2021** | | | | | | | | | |  |
| Programme | | : | **B.Tech – Computer Science & Engineering** | | Semester | | : | **Interim 2021-2022** | | |
| Course | | : | **Data Structures and Algorithms** | | Code | | : | **CSE2002** | | |
| Faculty | | : | **Dr. S. Rajasoundaran** | | Slot/Class No. | | : | **B11+DB1/0060** | | |
| Time | | : | **1½ hours** | | Max. Marks | | : | **50** | | |
| **Answer all the Questions** | | | | | | | | | | |
| **Q. No.** | **Question Description** | | | | | | | | **Marks** | |
| 1 | Write separate programs for finding the factorial value for a given number using recursive function and without using recursive function. Analyse the working strategies of both programs in terms of time complexity and space complexity. Compare and justify the best program with necessary steps. | | | | | | | | **10** | |
| 2 | State the conditions of different asymptotic notations for representing time complexities. Analyse the time complexity boundaries for the following functions.  Set your own boundary conditions (choose any . Check the possibilities for complexities. | | | | | | | | **10** | |
| 3 | Write sequential search and binary search algorithms. Implement these search techniques for finding the element “a” (Use any programming language). Compare the time complexities of both searching techniques under best case, worst case and average case. Which is the best technique? Justify with proper example. | | | | | | | | **10** | |
| 4 | What is the working principle of Divide and Conquer algorithms? Give one detailed scenario with proper negotiations for Divide and Conquer based sorting technique. Implement the same sorting technique using any programming language. | | | | | | | | **10** | |
| 5 | What are the ADTs and operations of stack data structure? Illustrate the operations of stack with proper diagrams. In addition, write the program for implementing the stack using arrays for the following scenario.   * Create new stack, size(stack)=5 * Push data for five times in to new stack * Pop data for three times from the stack * Illustrate stack overflow and underflow states | | | | | | | | **10** | |
| ⇔⇔⇔ | | | | | | | | | | |