**UE17CS202: Data Structures**

# of Credits: 4 # of Hrs: 52

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CLASS**  **#** | CHAPTER TITLE / REFERENCE  **LITERATURE** | TOPIC TO BE COVERED | % OF PORTIONS COVERED | |
| **REFERENCE CHAPTER** | CUMULATIVE |
| **Unit #: 1 – Introduction/Lists** | | | | |
|  | **T1: 1,4.7,4.8,5, Appendix**  **T2: 3,4** | Overview of course, Programming Practices | 19 | 19 |
|  | Pointers, Recursion |
|  | List Definition and array implementations |
|  | Link List operations – Insert, delete, traverse |
|  | Position based operations. |
|  | Link List operations – concatenate, merge |
|  | Link List operations – reverse etc |
|  | DLL – Definition, Linked List Implementation |
|  | DLL – operations, Header/Trailer nodes |
|  | Circular List, Multi List, Applications |
| **Unit #: 2 Stacks/Queues** | | | | |
|  | **T1: 3,12**  **T2 : 2,4** | Stack – Definition and Array Implementation | 19 | 38 |
|  | Stack – Linked List Implementation |
|  | Stack – Applications (post fix conversion) |
|  | Stack – Applications (expression evaluation) |
|  | Stack – parentheses balancing etc. |
|  | Queue – Definition and Operations |
|  | Queue – LL/Array implementation |
|  | Circular Queues – Implementation |
|  | Double ended Queue |
|  | Review of units |
| **Unit #:3 Graphs, Trees** | | | | |
|  | **T1: 4.1-4.5**  **T2: 5.1,5.2** | Graphs definitions and concepts | 20 | 58 |
|  | Complete, Regular graphs, connectivity |
|  | Euler and Hamilton Graphs |
|  | Representation of Graphs |
|  | Traversal of Graphs : BFS, DFS |
|  | Traversal of Graphs : BFS, DFS |
|  | Trees : representation, traversals, applications |
|  | Binary Trees Definition and properties |
|  | Binary Tree Implementation |
|  | Binary Trees Recursive methods/Traversals |
| **Unit #:4 Search Trees, Heap Trees** | | | | |
|  | **T1: 9.1,9.3,9.4**  **T2: 7.2** | Building and Evaluating binary Expression Tree | 19 | 77 |
|  | BST- Definition, Applications |
|  | BST- Implementation |
|  | BST – Search and Insert Operations |
|  | BST – Delete Operation |
|  | Heap Trees – Implementation |
|  | Heap Trees – FindMin, Insert |
|  | Heap Trees – Delete |
|  | Priority Queue using Heap |
|  | Review of units |
| **Unit #: 5 Tries/Hash Table** | | | | |
|  | **T1: 8.6,10.3**  **T2 : 7.3,7.4** | Tries : Definition | 23 | 100 |
|  | Tries : Implementation |
|  | Tries : Applications |
|  | Hash Table |
|  | Hash Functions |
|  | Collision Handling – Separate Chaining |
|  | Collision Handling – Separate Chaining |
|  | Collision Handling – Open Addressing |
|  | Collision Handling – Open Addressing |
|  | Hash Applications |
|  | Comparison of Data Structures |
|  | Course Summary and Conclusion |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Book Type** | **Code** | **Title & Author** | ***Publication Info*** | | |
| **Edition** | **Publisher** | **Year** |
| Text Book | T1 | Robert Kruse, C.L.Tondo, Bruce Leung and Shashi Mogalla, *“Data Structures and Program Design in C”,* Second Edition, Pearson/PHI, 2015 | 2 | Pearson/PHI | 2015 |
| Reference | T2 | Tanenbaum, Langsam, Augenstein “*Data Structures Using C and C++”*, Pearson/Prentice Hall , 2nd Edition, 2015 | 2 | Pearson/Prentice Hall | 2015 |