FILE STRUCTURES
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2016 -2017)
SEMESTER – VI

Subject Code	15IS62	IA Marks	20
Number of Lecture Hours/Week	4	Exam Marks	80
Total Number of Lecture Hours	50	Exam Hours	03

CREDITS – 04

Course objectives: This course will enable students to

- Explain the fundamentals of file structures and their management.
- Measure the performance of different file structures
- Organize different file structures in the memory.
- Demonstrate hashing and indexing techniques.

Module – 1	**Teaching Hours**
Introduction: File Structures: The Heart of the file structure Design, A Short History of File Structure Design, A Conceptual Toolkit; Fundamental File Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters, The Unix Directory Structure, Physical devices and Logical Files, File-related Header Files, UNIX file System Commands; Secondary Storage and System Software: Disks, Magnetic Tape, Disk versus Tape; CD-ROM: Introduction, Physical Organization, Strengths and Weaknesses; Storage as Hierarchy, A journey of a Byte, Buffer Management, Input /Output in UNIX. **Fundamental File Structure Concepts, Managing Files of Records :** Field and Record Organization, Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Managing Fixed Length, Fixed Field Buffers, An Object-Oriented Class for Record Files, Record Access, More about Record Structures, Encapsulating Record Operations in a Single Class, File Access and File Organization.	**10 Hours**
Module – 2	
Organization of Files for Performance, Indexing: Data Compression, Reclaiming Space in files, Internal Sorting and Binary Searching, Keysorting; What is an Index? A Simple Index for Entry-Sequenced File, Using Template Classes in C++ for Object I/O, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, Indexes that are too large to hold in Memory, Indexing to provide access by Multiple keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index structure: Inverted Lists, Selective indexes, Binding.	**10 Hours**
Module – 3	
Consequential Processing and the Sorting of Large Files: A Model for Implementing Cosequential Processes, Application of the Model to a General Ledger Program, Extension of the Model to include Mutiway Merging, A Second Look at Sorting in Memory, Merging as a Way of Sorting Large Files on Disk. **Multi-Level Indexing and B-Trees:** The invention of B-Tree, Statement of the problem, Indexing with Binary Search Trees; Multi-Level Indexing, B-Trees, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods; Nomenclature, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging and Redistribution, Redistribution during	**10 Hours**

insertion; B* Trees, Buffering of pages; Virtual B-Trees; Variable-length Records and keys.	
Module – 4	
Indexed Sequential File Access and Prefix B + Trees: Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree and its maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order B- Tree, Loading a Simple Prefix B+ Trees, B-Trees, B+ Trees and Simple Prefix B+ Trees in Perspective.	**10 Hours**
Module – 5	
Hashing: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distribution, How much Extra Memory should be used?, Collision resolution by progressive overflow, Buckets, Making deletions, Other collision resolution techniques, Patterns of record access. **Extendible Hashing:** How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches.	**10 Hours**
Course outcomes: The students should be able to:	
• Choose appropriate file structure for storage representation. • Identify a suitable sorting technique to arrange the data. • Select suitable indexing and hashing techniques for better performance to a given problem.	
Question paper pattern: The question paper will have TEN questions. There will be TWO questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer FIVE full questions, selecting ONE full question from each module.	
Text Books:	
1. Michael J. Folk, Bill Zoellick, Greg Riccardi: File Structures-An Object Oriented Approach with C++, 3rd Edition, Pearson Education, 1998. (Chapters 1 to 12 excluding 1.4, 1.5, 5.5, 5.6, 8.6, 8.7, 8.8)	
Reference Books:	
1. K.R. Venugopal, K.G. Srinivas, P.M. Krishnaraj: File Structures Using C++, Tata McGraw-Hill, 2008. 2. Scot Robert Ladd: C++ Components and Algorithms, BPB Publications, 1993. 3. Raghu Ramakrishan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw Hill, 2003.	