# **Core Programming**

* Programming in C – A Complete Introduction to The C Programming Language, Stephen G. Kochan
* Object-Oriented Programming with C++, E Balagurusamy

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Object Oriented Programming with C++** |  | **Programming in C** |
|  | **Principles of Object-Oriented Programming** |  | **Introduction and Some Fundamentals** |
| 01 | Software Crisis | 01 | Programming |
| 02 | Software Evolution | 02 | Higher-Level Languages |
| 03 | A Look at Procedure-Oriented Program | 03 | Integrated Development Environments |
| 04 | Object-Oriented Programming Paradigm | 04 | Operating System |
| 05 | Basic Concepts of Object-Oriented Programming | 05 | Compiling Programs |
| 06 | Benefits of OOP | 06 | Language Interpreters |
| 07 | Object-Oriented Languages |  | **Compiling and Running Your First Program** |
| 08 | Applications of OOP | 07 | Compiling Your Program |
|  | **Beginning with C++** | 08 | Running Your Program |
| 09 | What is C++ | 09 | Understanding Your First Program |
| 10 | Applications of C++ | 10 | Displaying the Values of Variables |
| 11 | A Simple C++ Program | 11 | Comments |
| 12 | More C++ Statements |  | **Variables, Data Types, and Arithmetic Expressions** |
| 13 | An Example with Class | 12 | Working with Variables |
| 14 | Structure of C++ Program | 13 | Understanding Data Types and Constants |
| 15 | Creating the Source File | 14 | Working with Arithmetic Expression |
| 16 | Compiling and Linking | 15 | The Assignment Operators |
|  | **Tokens, Expression and Control Structures** | 16 | Types \_Complex and \_Imaginary |
| 17 | Introduction |  | **Program Looping** |
| 18 | Tokens | 17 | The (for) Statement and Relational Operators |
| 19 | Keywords | 18 | The (While) Statement |
| 20 | Identifiers and Constants | 19 | The (do) Statement |
| 21 | Basic Data Types | 20 | The (break) Statement |
| 22 | User-Defined Data Types | 21 | The (continue) Statement |
| 23 | Storage Classes |  | **Making Decisions** |
| 24 | Derived Data Types | 22 | The (if) Statement |
| 25 | Symbolic Constants | 23 | The (switch) Statement |
| 26 | Type Compatibility | 24 | Boolean Variables |
| 27 | Declaration of Variables | 25 | The Conditional Operators |
| 28 | Dynamic Initialization of Variables |  | **Working with Arrays** |
| 29 | Reference Variables | 26 | Defining an Array |
| 30 | Operators in C++ | 27 | Initializing Arrays |
| 31 | Scope Resolution Operator | 28 | Character Arrays |
| 32 | Member Dereferencing Operators | 29 | Multidimensional Arrays |
| 33 | Memory Management Operators | 30 | Variable-Length Arrays |
| 34 | Manipulators |  | **Working with Functions** |
| 35 | Type Cast Operator | 31 | Defining a Function |
| 36 | Expressions and Their Types | 32 | Arguments and Local Variables |
| 37 | Special Assignment Expressions | 33 | Returning Function Results |
| 38 | Implicit Conversions | 34 | Functions Calling Functions Calling |
| 39 | Operator Overloading | 35 | Top-Down Programming |
| 40 | Operator Precedence | 36 | Functions and Arrays |
| 41 | Control Structures | 37 | Global Variables |
|  | **Functions in C++** | 38 | Automatic and Static Variables |
| 42 | Introduction | 39 | Recursive Functions |
| 43 | The Main Function |  | **Working with Structures** |
| 44 | Function Prototyping | 40 | A Structure for Storing the Date |
| 45 | Call by Reference | 41 | Functions and Structures |
| 46 | Inline Functions | 42 | Initializing Structures |
| 47 | Default Arguments | 43 | Arrays of Structures |
| 48 | Const Arguments | 44 | Structures Containing Structures |
| 49 | Recursion | 45 | Structures Containing Arrays |
| 50 | Function Overloading | 46 | Structure Variants |
| 51 | Friend and Virtual Functions |  | **Character Strings** |
| 52 | Math Library Functions | 47 | Arrays of Characters |
|  | **Classes and Objects** | 48 | Variable-Length Character Strings |
| 53 | Introduction | 49 | Escape Characters |
| 54 | C Structures Revisited | 50 | More on Constant Strings |
| 55 | Specifying a Class 92 | 51 | Character Strings, Structures, and Arrays |
| 54 | Defining Member Functions | 52 | Character Operations |
| 55 | A C++ Program with Class |  | **Pointers** |
| 56 | Making an Outside Function Inline | 53 | Defining a Pointer Variable |
| 57 | Nesting of Member Functions | 54 | Using Pointers in Expressions |
| 58 | Private Member Functions | 55 | Working with Pointers and Structures |
| 59 | Arrays within a Class | 56 | The Keyword (Const) and Pointers |
| 60 | Memory Allocation for Objects | 57 | Pointers and Functions |
| 61 | Static Data Members | 58 | Pointers and Arrays |
| 62 | Static Member Functions | 59 | Operations on Pointers |
| 63 | Arrays of Objects | 60 | Pointers to Functions |
| 64 | Objects as Function Arguments | 61 | Pointers and Memory Addresses |
| 65 | Friendly Functions |  | **Operations on Bits** |
| 66 | Returning Objects | 62 | Bit Operators |
| 67 | (const) Member Functions | 63 | Bit Fields |
| 68 | Pointers to Members |  | **The Preprocessor** |
| 69 | Local Classes | 64 | The (#define) Statement |
|  | **Constructors and Destructors** | 65 | The (#include) Statement |
| 70 | Introduction | 66 | Conditional Compilation |
| 71 | Constructors |  | **More on Data Types** |
| 72 | Parameterized Constructors | 67 | Enumerated Data Types |
| 73 | Multiple Constructors in a Class | 68 | The (typedef) Statement |
| 74 | Constructors with Default Arguments | 69 | Data Type Conversions |
| 75 | Dynamic Initialization of Objects |  | **Working with Larger Programs** |
| 76 | Copy Constructor | 70 | Dividing Your Program into Multiple Files |
| 77 | Dynamic Constructors | 71 | Communication Between Modules |
| 78 | Constructing Two-Dimensional Arrays | 72 | Other Utilities for Working with Larger |
| 79 | (const) Objects | 73 | Programs |
| 80 | Destructors |  | **Input and Output Operations in C** |
|  | **Operator Overloading and Type Conversions** | 74 | Character I/O: getcher and putchar |
| 81 | Introduction | 75 | Formatted I/O: printf and scanf |
| 82 | Defining Operator Overloading | 76 | Input and Output Operations with Files |
| 83 | Overloading Unary Operators | 77 | Special Functions for Working with Files |
| 84 | Overloading Binary Operators |  | **Miscellaneous and Advanced Features** |
| 85 | Overloading Binary Operators Using Friends | 78 | Miscellaneous Language Statements |
| 86 | Manipulation of Strings Using Operators | 79 | Working with Unions |
| 87 | Some Other Operator Overloading Examples | 80 | The Comma Operator |
| 88 | Rules for Overloading Operators | 81 | Type Qualifiers |
| 89 | Type Conversions | 82 | Command-Line Arguments |
|  | **Inheritance: Extending Classes** | 83 | Dynamic Memory Allocation |
| 90 | Introduction |  | **Debugging Programs** |
| 91 | Defining Derived Classes | 84 | Debugging with the Preprocessor |
| 92 | Single Inheritance | 85 | Debugging Programs with gdb |
| 93 | Making a Private Member Inheritable | 86 | Object-Oriented Programming |
| 94 | Multilevel Inheritance | 87 | What is an Object Anyway |
| 95 | Multiple Inheritance | 88 | Instances and Methods |
| 96 | Hierarchical Inheritance | 89 | Writing a C Program to Work with Fractions |
| 97 | Hybrid Inheritance | 90 | Defining an Objective-C Class to Work with Fractions |
| 98 | Virtual Base Classes | 91 | Defining a C++ Class to Work with Fractiosn |
| 99 | Abstract Classes | 92 | Defining a C# Class to Work with Fractions |
| 100 | Constructors in Derived Classes |  | **The Standard C Library** |
| 101 | Member Classes: Nesting of Classes | 93 | Standard Header Files |
|  | **Pointers, Virtual Functions and Polymorphism** | 94 | String Functions |
| 102 | Introduction | 95 | Memory Functions |
| 103 | Pointers | 96 | Character Functions |
| 104 | Pointers to Objects | 97 | I/O Functions |
| 105 | (this) Pointer | 98 | In-Memory Format Conversion Functions |
| 106 | Pointers to Derived Classes | 99 | String-to-Number Conversion |
| 107 | Virtual Functions | 100 | Dynamic Memory Allocation Functions |
| 108 | Pure Virtual Functions | 101 | Math Functions |
| 109 | Virtual Constructors and Destructors | 102 | General Utility Functions |
|  | **Managing Console I/O Operations** |  | **Compiling Programs with gcc** |
| 110 | Introduction | 103 | General Command Format |
| 111 | C++ Streams | 104 | Command-Line Options |
| 112 | C++ Stream Classes |  | **Common Programming Mistakes** |
| 113 | Unformatted I/O Operations |  |  |
| 114 | Formatted Console I/O Operations |  |  |
| 115 | Managing Output with Manipulators |  |  |
|  | **Working with Files** |  |  |
| 116 | Introduction |  |  |
| 117 | Classes for File Stream Operations |  |  |
| 118 | Opening and Closing a File |  |  |
| 119 | Detecting end-of-file |  |  |
| 120 | More about Open(): File Modes |  |  |
| 121 | File Pointers and their Manipulations |  |  |
| 122 | Sequential Input and Output Operations |  |  |
| 123 | Updating a File: Random Access |  |  |
| 124 | Error Handling During File Operations |  |  |
| 125 | Command-line Arguments |  |  |
|  | **Templates** |  |  |
| 126 | Introduction |  |  |
| 127 | Class Templates |  |  |
| 128 | Class Templates with Multiple Parameters |  |  |
| 129 | Function Templates |  |  |
| 130 | Function Templates with Multiple Parameters |  |  |
| 131 | Overloading of Template Functions |  |  |
| 132 | Member Function Templates |  |  |
| 133 | Nontype Template Arguments |  |  |
|  | **Exception Handling** |  |  |
| 134 | Introduction |  |  |
| 135 | Basics of Exception Handling |  |  |
| 136 | Exception Handling Mechanism |  |  |
| 137 | Throwing Mechanism |  |  |
| 138 | Catching Mechanism |  |  |
| 139 | Re-throwing an Exception |  |  |
| 140 | Specifying Exceptions |  |  |
| 141 | Exceptions in Constructors and Destructors |  |  |
| 142 | Exceptions in Operator Overloaded Functions |  |  |
|  | **Introduction to the Standard Template Library** |  |  |
| 143 | Introduction |  |  |
| 144 | Components of STL |  |  |
| 145 | Containers |  |  |
| 146 | Algorithms |  |  |
| 147 | Iterators |  |  |
| 148 | Applications |  |  |
| 149 | Of Container Classes |  |  |
| 150 | Function Objects |  |  |
|  | **Manipulating Strings** |  |  |
| 151 | Introduction |  |  |
| 152 | Creating (string) Objects |  |  |
| 153 | Manipulating String Objects |  |  |
| 154 | Relational Operations |  |  |
| 155 | String Characteristics |  |  |
| 156 | Accessing Characters in Strings |  |  |
| 157 | Comparing and Swapping |  |  |
|  | **New Features of ANSI C++ Standard** |  |  |
| 158 | Introduction |  |  |
| 159 | New Data Types |  |  |
| 160 | New Operators |  |  |
| 161 | Class Implementation |  |  |
| 162 | Namespace Scope |  |  |
| 163 | Operator Keywords |  |  |
| 164 | New Keywords |  |  |
| 165 | New Headers |  |  |
|  | **Object-Oriented Systems Development** |  |  |
| 166 | Introduction |  |  |
| 167 | Procedure-Oriented Paradigm |  |  |
| 168 | Procedure-Oriented Development Tools |  |  |
| 169 | Object-Oriented Paradigm |  |  |
| 170 | Object-Oriented Notations and Graphs |  |  |
| 171 | Steps in Object-Oriented Analysis |  |  |
| 172 | Steps in Object-Oriented Design |  |  |

# **Database Management System**

* Fundamentals of Database Systems, Elmasri and Navathe

|  |  |
| --- | --- |
|  | **Fundamental of Database System** |
|  | **Part-1: Introduction to Database** |
|  | **Databases and Database Users** |
| 01 | Introduction |
| 02 | An Example |
| 03 | Characteristics of the Database Approach |
| 04 | Actors on the Scene |
| 05 | Workers behind the Scene |
| 06 | Advantages of Using the DBMS Approach |
| 07 | A Brief History of Database Applications |
| 08 | When Not to Use a DBMS |
|  | **Database System Concepts and Architecture** |
| 09 | Data Models, Schemas, and Instances |
| 10 | Three-Schema Architecture and Data Independence |
| 11 | Database Languages and Interfaces |
| 12 | The Database System Environment |
| 13 | Centralized and Client/Server Architectures for DBMS |
| 14 | Classification of Database Management Systems |
|  | **Part-2: The Relational Data Model and SQL** |
|  | **The Relational Data Model and Database Constraints** |
| 15 | Relational Model Concepts |
| 16 | Relational Model Constraints and Relational Database Schemas |
| 17 | Update Operations, Transactions, and Dealing with Constraint Violations |
|  | **Basic SQL** |
| 18 | SQL Data Definition and Data Types |
| 19 | Specifying Constraints in SQL |
| 20 | Basic Retrieval Queries in SQL |
| 21 | INSERT, DELETE, and UPDATE Statements in SQL |
| 22 | Additional Features of SQL |
|  | **More SQL: Complex Queries, Triggers, and Schema Modification** |
| 23 | More Complex SQL Retrieval Queries |
| 24 | Specifying Constraints as Assertions and Actions as Triggers |
| 25 | Views (Virtual Tables) in SQL |
| 26 | Schema Change Statements in SQL |
|  | **The Relational Algebra and Relational Calculus** |
| 27 | Unary Relational Operations: SELECT and PROJECT |
| 28 | Relational Algebra Operations from Set Theory |
| 29 | Binary Relational Operations: JOIN and DIVISION |
| 30 | Additional Relational Operations |
| 31 | Examples of Queries in Relational Algebra |
| 32 | The Tuple Relational Calculus |
| 33 | The Domain Relational Calculus |
|  | **Part-03: Conceptual Modeling and Database Design** |
|  | **Data Modeling Using the Entity-Relationship (ER) Model** |
| 34 | Using High-Level Conceptual Data Models for Database Design |
| 35 | A Sample Database Application |
| 36 | Entity Types, Entity Sets, Attributes, and Keys |
| 37 | Relationship Types, Relationship Sets, Roles, and Structural Constraints |
| 38 | Weak Entity Types |
| 39 | Refining the ER Design for the COMPANY Database |
| 40 | ER Diagrams, Naming Conversions, and Design Issues |
| 41 | Example of Other Notation: UML Class Diagrams |
| 42 | Relationship Types of Degree Higher than Two |
|  | **The Enhanced Entity-Relationship (EER) Model** |
| 43 | Sub-classes, Super-classes, and Inheritance |
| 44 | Specialization and Generalization |
| 45 | Constraints and Characteristics of Specialization and Generalization Hierarchies |
| 46 | Modeling of UNION Types Using Categories |
| 47 | A Simple UNIVERSITY EER Schema, Design Choices, and Formal Definitions |
| 48 | Example of Other Notation: Representing Specialization and Generalization in UML Class Diagram |
| 49 | Data Abstraction, Knowledge Representation, and Ontology Concepts |
|  | **Relational Database Design by ER- and EER-to-Relational Mapping** |
| 50 | Relational Database Design Using ER-to-Relational Mapping |
| 51 | Mapping EER Model Constructs to Relations |
|  | **Practical Database Design Methodology and Use of UML Diagrams** |
| 52 | The Role of Information Systems in Organizations |
| 53 | The Database Design and Implementation Process |
| 54 | Use of UML Diagrams as an Aid to Database Design Specification |
| 55 | Retional Rose: A UML-Based Design Tool |
| 56 | Automated Database Design Tools |
|  | **Part-4: Object, Object-Retational, and XML: Concepts, Models, Languages, and Standards** |
|  | **Object and Object-Relational Database** |
| 57 | Overview of Object Database Concepts |
| 58 | Object-Relational Features: Object Database Extensions to SQL |
| 59 | The ODMG Object Model and the Object Definition Language ODL |
| 60 | Object Database Conceptual Design |
| 61 | The Object Query Language OQL |
| 62 | Overview of the C++ Language Binding in the ODMG Standard |
|  | **XML: Extensible Markup Language** |
| 63 | Structured, Semi structured, and Unstructured Data |
| 64 | XML Hierarchical (Tree) Data Model |
| 65 | XML Documents, DTD, and XML Schema |
| 66 | Storing and Extracting XML Documents from Databases |
| 67 | XML Languages |
| 68 | Extracting XML Documents from Relational Databases |
|  | **Part-5: Database Programming Techniques** |
|  | **Introduction to SQL Programming Techniques** |
| 69 | Database Programming: Techniques and Issues |
| 70 | Embedded SQL, dynamic SQL, and SQLI |
| 71 | Database Programming with Function Calls: SQL/CLI and HDBC |
| 72 | Database Stored Procedures and SQL/PSM |
| 73 | Comparing the Three Approaches |
|  | **Web Database Programming Using PHP** |
| 74 | A Simple PHP Example |
| 75 | Overview of Basic Features of PHP |
| 76 | Overview of PHP Database Programming |
|  | **Part-6: Database Design Theory and Normalization** |
|  | **Basics of Functional Dependencies and Normalization for Relational Databases** |
| 77 | Information Design Guidelines for Relation Schemas |
| 78 | Functional Dependencies |
| 79 | Normal Forms Based on Primary Keys |
| 80 | General Definitions of Second and Third Normal Forms |
| 81 | Boyce-Codd Normal Form |
| 82 | Multivalued Dependency and Fourth Normal Form |
| 83 | Join Dependencies and Fifth Normal Form |
|  | **Relational Database Design Algorithms and Further Dependencies** |
| 84 | Further Topics in Functional Dependencies: Inference Rules, Equivalence, and Minimal Cover |
| 85 | Properties of Relational Decompositions |
| 86 | Algorithms for Relational Database Schema Design |
| 87 | About Nulls, Dangling Tuples, and Alternative Relational Designs |
| 88 | Further Discussion of Multivalued Dependencies and 4NF |
| 89 | Other Dependencies and Normal Forms |
|  | **Part-7: File Structures, Indexing, and Hashing** |
|  | **Disk Storage, Basic File Structures, and Hashing** |
| 90 | Introduction |
| 91 | Secondary Storage Devices |
| 92 | Buffering of Blocks |
| 93 | Placing File Records on Disk |
| 94 | Operations on Files |
| 95 | Files of Unordered Records (Heap Files) |
| 96 | Files of Ordered Records (Sorted Files) |
| 97 | Hashing Techniques |
| 98 | Other Primary File Organizations |
| 99 | Parallelizing Disk Access Using RAID Technologies |
| 100 | New Storage Systems |
|  | **Indexing Structures for Files** |
| 101 | Types of Single-Level Ordered Indexes |
| 102 | Multilevel Indexes |
| 103 | Dynamic Multilevel Indexes Using B-Trees and B+-Trees |
| 104 | Indexes on Multiple Keys |
| 105 | Other Types of Indexes |
| 106 | Some General Issues Concerning Indexing |
|  | **Part-8: Query Processing and Optimization, and Database Tuning** |
|  | **Algorithms for Query Processing and Optimization** |
| 107 | Translating SQL Queries into Relational Algebra |
| 108 | Algorithms for External Sorting |
| 109 | Algorithms for SELECT and JOIN Operations |
| 110 | Algorithms for PROJECT and Set Operations |
| 111 | Implementing Aggregate Operations and OUTER JOINs |
| 112 | Combining Operations Using Pipelining |
| 113 | Using Heuristics in Query Optimization |
| 114 | Using Selectivity and Cost Estimates in Query Optimization |
| 115 | Overview of Query Optimization in Oracle |
| 116 | Semantic Query Optimization |
|  | **Physical Database Design and Tuning** |
| 117 | Physical Database Design in Relational Databases |
| 118 | An Overview of Database Tuning in Relational Systems |
|  | **Part-9: Transaction Processing, Concurrency Control, and Recovery** |
|  | **Introduction to Transaction Processing Concepts and Theory** |
| 119 | Introduction to Transaction Processing |
| 120 | Transaction and System Concepts |
| 121 | Desirable Properties of Transactions |
| 122 | Characterizing Schedules Based on Recoverability |
| 123 | Characterizing Schedules Based on Serializability |
| 124 | Transaction Support in SQL |
|  | **Concurrency Control Techniques** |
| 125 | Two-Phase Locking Techniques for Concurrency Control |
| 126 | Concurrency Control Based on Timestamp Ordering |
| 127 | Multi-version Concurrency Control Techniques |
| 128 | Validation (Optimistic) Concurrency Control Techniques |
| 129 | Granularity of Data Items and Multiple Granularity Locking |
| 130 | Using Locks for Concurrency Control in Indexes |
| 131 | Other Concurrency Control Issues |
|  | **Database Recovery Techniques** |
| 132 | Recovery Concepts |
| 133 | NO-UNDO/REDO Recovery Based on Deferred Update |
| 134 | Recovery Techniques Based on Immediate Update |
| 135 | Shadow Paging |
| 136 | The ARIES Recovery Algorithm |
| 137 | Recovery in Multi-database Systems |
| 138 | Database Backup and Recovery from Catastrophic Failures |
|  | **Part-10: Additional Database Topics: Security and Distribution** |
|  | **Database Security** |
| 139 | Introduction to Database Security Issues |
| 140 | Discretionary Access Control Based on Granting and Revoking Privileges |
| 141 | Mandatory Access Control and Role-Based Access Control for Multilevel Security |
| 142 | SQL Injection |
| 143 | Introduction to Statistical Database Security |
| 144 | Introduction to Flow Control |
| 145 | Encryption and Public Key Infrastructures |
| 146 | Privacy Issues and Preservation |
| 147 | Challenges of Database Security |
| 148 | Oracle Label-Based Security |
|  | **Distributed Databases** |
| 149 | Distributed Database Concepts |
| 150 | Types of Distributed Database Systems |
| 151 | Distributed Database Architectures |
| 152 | Data Fragmentation, Replication, and Allocation Techniques for Distributed Database Design |
| 153 | Query Processing and Optimization in Distributed Databases |
| 154 | Overview of Transaction Management in Distributed Databases |
| 155 | Overview of Concurrency Control and Recovery in Distributed Databases |
| 156 | Distributed Catalog Management |
| 157 | Current Trends in Distributed Databases |
|  | **Distributed Databases in Oracle** |
|  | **Part-11: Advanced Database Models, Systems, and Applications** |
| 158 | Active Database Concepts and Triggers |
| 159 | Temporal Database Concepts |
| 160 | Spatial Database Concepts |
| 161 | Multimedia Database Concepts |
| 162 | Introduction to Deductive Databases |
|  | **Introduction to Information and Web Search** |
| 163 | Information Retrieval (IR) Concepts |
| 164 | Retrieval Models |
| 165 | Types of Queries in IR Systems |
| 166 | Text Preprocessing |
| 167 | Inverted Indexing |
| 168 | Evaluation Measures of Search Relevance |
| 169 | Web Search and Analysis |
| 170 | Trends in Information Retrieval |
|  | **Data Mining Concepts** |
| 171 | Overview of Data Mining Technology |
| 172 | Association Rules |
| 173 | Classification |
| 174 | Clustering |
| 175 | Approaches to Other Data Mining Problems |
| 176 | Applications of Data Mining |
| 177 | Commercial Data Mining Tools |
|  | **Overview of Data Warehouse and OLAP** |
| 178 | Introduction, Definitions, and Terminology |
| 179 | Characteristics of Data Warehouses |
| 180 | Data modeling for Data Warehouses |
| 181 | Building a Data Warehouse |
| 182 | Typical Functionality of a Data warehouse |
| 183 | Data Warehouse versus Views |
| 184 | Difficulties of Implementing Data Warehouse |
|  | **Appendix** |
| 185 | Alternative Diagrammatic Notations for ER Models |
| 186 | Parameters of Disks |
| 187 | Overview of the OBE Language |
| 188 | Overview of the Hierarchical Data Model |
| 189 | Overview of the Network Data Model |

# **Data Structure and Algorithm**

* Classic Data Structures, D. Samanta
* Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald, Clifford

|  |  |
| --- | --- |
|  | **Classic Data Structure** |
|  | **Introduction and Overview** |
| 01 | Definitions |
| 02 | Concept of Data Structures |
| 03 | Overview of Data Structures |
| 04 | Implementation of Data Structures |
|  | **Arrays** |
| 05 | Definition |
| 06 | Terminology |
| 07 | On-Dimensional Array |
| 08 | Multidimensional Arrays |
| 09 | Pointer Arrays |
|  | **Linked Lists** |
| 10 | Definition |
| 11 | Single Linked List |
| 12 | Circular Linked List |
| 13 | Double Linked Lists |
| 14 | Circular Double Linked List |
| 15 | Application of Linked Lists |
| 16 | Memory Representation |
| 17 | Boundary Tag System |
| 18 | Deallocation Strategy |
| 19 | Buddy System |
| 20 | Compaction |
|  | **Stacks** |
| 21 | Introduction |
| 22 | Definition |
| 23 | Representation of Stack |
| 24 | Operations on Stacks |
| 25 | Applications of Stacks |
|  | **Queues** |
| 26 | Introduction |
| 27 | Definition |
| 28 | Representation of Queues |
| 29 | Various Queue Structures |
| 30 | Application of Queue |
|  | **Tables** |
| 31 | Rectangular Tables |
| 32 | Jagged Tables |
| 33 | Inverted Tables |
| 34 | Hash Tables |
|  | **Trees** |
| 35 | Basic Terminologies |
| 36 | Definition and Concepts |
| 37 | Representation of Binary Tree |
| 38 | Operations on Binary Tree |
| 39 | Types of Binary Trees |
| 40 | Trees and Forests |
| 41 | Trees |
| 42 | B+ Tree Indexing |
| 43 | Trie Tree Indexing |
|  | **Graphs** |
| 44 | Introduction |
| 45 | Graph Terminologies |
| 46 | Representation of Graphs |
| 47 | Operations on Graphs |
| 48 | Application of Graph Structures |
| 49 | BDD and its Applications |
|  | **Sets** |
| 50 | Definition and Terminologies |
| 51 | Representation of Sets |
| 52 | Operations of Sets |
| 53 | Applications of Sets |
|  | **Algorithm** |
|  | **Introduction to Algorithm** |
|  | **Part-1: Foundations** |
| 54 | The Role of Algorithms in Computing |
| 55 | Getting Started |
| 56 | Growth of Functions |
| 57 | Divide-and-Conquer |
| 58 | Probabilistic Analysis and Randomized Algorithms |
|  | **Part-2: Sorting and Order Statistic** |
| 59 | Heapsort |
| 60 | Quicksort |
| 61 | Sorting in Linear Time |
| 62 | Medians and Order Statistics |
|  | **Part-3: Data Structures** |
| 63 | Elementary Data Structures |
| 64 | Hash Tables |
| 65 | Binary Search Trees |
| 66 | Red-Black Trees |
| 67 | Augmenting Data Structures |
|  | **Part-4: Advanced Design and Analysis Techniques** |
| 68 | Dynamic Programming |
| 69 | Greedy Algorithms |
| 70 | Amortized Analysis |
|  | **Part-5: Advanced Data Structure** |
| 71 | B-Trees |
| 72 | Fibonacci Heaps |
| 73 | Van Emde Boas Trees |
| 74 | Data Structures for Disjoint Sets |
|  | **Part-6: Graph Algorithms** |
| 75 | Elementary Graph Algorithms |
| 76 | Minimum Spanning Trees |
| 77 | Single-Source Shortest Paths |
| 78 | All-Pairs Shortest Paths |
| 79 | Maximum Flow |
|  | **Part-7: Selected Topics** |
| 80 | Multithreaded Algorithm |
| 81 | Matrix Operations |
| 82 | Linear Programming |
| 83 | Polynomials and the FFT |
| 84 | Number-Theoretic Algorithms |
| 85 | String Matching |
| 86 | Computational Geometry |
| 87 | NP-Completeness |
| 88 | Approximation Algorithms |
|  | **Part-8: Appendix: Mathematical Background** |
| 89 | Summations |
| 90 | Sets, Etc |
| 91 | Counting and Probability |
| 92 | Matrices |

# **Software Engineering and Maintenance**

* Software Engineering: A Practitioner’s Approach, Roger S. Pressman
* Software Maintenance, GPT

|  |  |
| --- | --- |
|  | **Software Engineering: A Practitioner’s Approach** |
|  | **Software and Software Engineering** |
| 01 | The Nature of Software |
| 02 | The Unique nature of WebApps |
| 03 | Software Engineering |
| 04 | The Software Process |
| 05 | Software Engineering Practice |
| 06 | Software Myths |
| 07 | How It All Starts |
|  | **Part-1: The Software Process** |
|  | **Process Models** |
| 08 | A Generic Process Model |
| 09 | Process Assessment and Improvement |
| 10 | Prescriptive Process Models |
| 11 | Specialized Process Models |
| 12 | The Unified Process |
| 13 | Personal and Team Process Models |
| 14 | Process Technology |
| 15 | Product and Process |
| 16 | Summary |
|  | **Agile Development** |
| 17 | What is Agility |
| 18 | Agility and the Cost of Change |
| 19 | What Is an Agile Process |
| 20 | Extreme Programming (XP) |
| 21 | Other Agile Process Models |
| 22 | A Tool Set for the Agile Process |
| 23 | Summary |
|  | **Part-2: Modeling** |
|  | **Principles That Guide Practice** |
| 24 | Software Engineering Knowledge |
| 25 | Core Principles |
| 26 | Principles That Guide Each Framework Activity |
| 27 | Summary |
|  | **Understanding Requirements** |
| 28 | Requirements Engineering |
| 29 | Establishing the Groundwork |
| 30 | Eliciting Requirements |
| 31 | Developing Use Cases |
| 32 | Building the Requirements Model |
| 33 | Negotiating Requirements |
| 34 | Validating Requirements |
| 35 | Summary |
|  | **Requirements Modeling: Scenarios, Information, and Analysis Classes** |
| 36 | Requirements Analysis |
| 37 | Scenario-Based Modeling |
| 38 | UML Models That Supplement the Use Case |
| 39 | Data Modeling Concepts |
| 40 | Class-Based Modeling |
| 41 | Summary |
|  | **Requirements Modeling: Flow, Behavior, Patterns, and WebApps** |
| 42 | Requirements Modeling Strategies |
| 43 | Flow-Oriented Modeling |
| 44 | Creating a Behavioral Model |
| 45 | Patterns for Requirements Modeling |
| 46 | Requirements Modeling for WebApps |
| 47 | Summary |
|  | **Design Concepts** |
| 48 | Design within the Context of Software Engineering |
| 49 | The Design Process |
| 50 | Design Concepts |
| 51 | The Design Model |
| 52 | Summary |
|  | **Architectural Design** |
| 53 | Software Architecture |
| 54 | Architectural Genres |
| 55 | Architectural Styles |
| 56 | Architectural Design |
| 57 | Assessing Alternative Architectural Designs |
| 58 | Architectural Mapping Using Data Flow |
| 59 | Summary |
|  | **Component-Level Design** |
| 60 | What is a Component |
| 61 | Designing Class-Based Components |
| 62 | Conducting Component-Level Design |
| 63 | Component-Level Design for WebApps |
| 64 | Designing Traditional components |
| 65 | Component-Based Development |
| 66 | Summary |
|  | **User Interface Design** |
| 67 | The Golden Rules |
| 68 | User Interface Analysis and Design |
| 69 | Interface Analysis |
| 70 | Interface Design Steps |
| 71 | WebApp Interface Design |
| 72 | Design Evaluation |
| 73 | Summary |
|  | **Pattern-Based Design** |
| 74 | Design Patterns |
| 75 | Pattern-Based Software Design |
| 76 | Architectural Patterns |
| 77 | Component-Level Design Patterns |
| 78 | User Interface Design Patterns |
| 79 | WebApp Design Patterns |
| 80 | Summary |
|  | **WebApp Design** |
| 81 | WebApp Design Quality |
| 82 | Design Goals |
| 83 | A Design Pyramid for WebApps |
| 84 | Aesthetic Design |
| 85 | Content Design |
| 86 | Architecture Design |
| 87 | Navigation Design |
| 88 | Component-Level Design |
| 89 | Object-Oriented Hypermedia Design Method (OOHDM) |
| 90 | Summary |
|  | **Part-3: Quality Management** |
|  | **Quality Concepts** |
| 91 | What is Quality |
| 92 | Software Quality |
| 93 | The Software Quality Dilemma |
| 94 | Achieving Software Quality |
| 95 | Summary |
|  | **Review Techniques** |
| 96 | Cost Impact of Software Defects |
| 97 | Defect Amplification and Removal |
| 98 | Review Metrics and Their Use |
| 99 | Reviews: A formality Spectrum |
| 100 | Informal Reviews |
| 101 | Formal Technical Reviews |
| 102 | Summary |
|  | **Software Quality Assurance** |
| 103 | Background Issues |
| 104 | Elements of Software Quality Assurance |
| 105 | SQA Tasks, Goals, and Metrics |
| 106 | Formal Approaches to SQA |
| 107 | Statistical Software Quality Assurance |
| 108 | Software Reliability |
| 109 | The ISO 9000 Quality Standards |
| 110 | The SQA Plan |
| 111 | Summary |
|  | **Software Testing Strategies** |
| 112 | A Strategic Approach to Software Testing |
| 113 | Strategic Issues |
| 114 | Test Strategies for Conventional Software |
| 115 | Test Strategies for Object-Oriented Software |
| 116 | Test Strategies for WebApps |
| 117 | Validation Testing |
| 118 | System Testing |
| 119 | The Art of Debugging |
| 120 | Summary |
|  | **Testing Conventional Application** |
| 121 | Software Testing Fundamentals |
| 122 | Internal and External Views of Testing |
| 123 | White-Box Testing |
| 124 | Basis Path Testing |
| 125 | Control Structure Testing |
| 126 | Black-Box Testing |
| 127 | Model-Based Testing |
| 128 | Testing for Specialized Environments, Architectures, and Applications |
| 129 | Patterns for Software Testing |
| 130 | Summary |
|  | **Testing Object-Oriented Applications** |
| 131 | Broadening the View of Testing |
| 132 | Testing OOA and OOD Models |
| 133 | Object-Oriented Testing Strategies |
| 134 | Object-Oriented Testing Methods |
| 135 | Testing Methods Applicable at the Class Level |
| 136 | Interclass Test-Case Design |
| 137 | Summary |
|  | **Testing Web Applications** |
| 138 | Testing Concepts for WebApps |
| 139 | The Testing Process – An Overview |
| 140 | Content Testing |
| 141 | User Interface Testing |
| 142 | Component-Level Testing |
| 143 | Navigation Testing Configuration Testing |
| 144 | Security Testing |
| 145 | Performance Testing |
| 146 | Summary |
|  | **Formal Modeling and Verification** |
| 147 | The Cleanroom Strategy |
| 148 | Functional Specification |
| 149 | Cleanroom Design |
| 150 | Cleanroom Testing |
| 151 | Formal Methods Concepts |
| 152 | Applying Mathematical Notation for Formal Specification |
| 153 | Formal Specification Languages |
| 154 | Summary |
|  | **Software Configuration Management** |
| 155 | Software Configuration Management |
| 156 | The SCM Repository |
| 157 | The SCM Process |
| 158 | Configuration management for WebApps |
| 159 | Summary |
|  | **Product Metrics** |
| 160 | A Framework for Product Metrics |
| 161 | Metrics for the Requirements Model |
| 162 | Metrics for the Design Model |
| 163 | Design Metrics for WebApps |
| 164 | Metrics for Source Code |
| 165 | Metrics for Testing |
| 166 | Metrics for Maintenance |
| 167 | Summary |
|  | **Part-4: Managing Software Projects** |
|  | **Project Management Concepts** |
| 168 | The Management Spectrum |
| 169 | People |
| 170 | The Product |
| 171 | The Process |
| 172 | The Project |
| 173 | The W5HH Principle |
| 174 | Critical Practices |
| 175 | Summary |
|  | **Process and Project Metrics** |
| 176 | Metrics in the Process and Project Domains |
| 177 | Software Measurement |
| 178 | Metrics for Software Quality |
| 179 | Integrating Metrics within the Software Process |
| 180 | Metrics for Small Organizations |
| 181 | Establishing a Software Metrics Program |
| 182 | Summary |
|  | **Estimation for Software Projects** |
| 183 | Observations on Estimation |
| 184 | The Project Planning Process |
| 185 | Software Scope and Feasibility |
| 186 | Resources |
| 187 | Software Project Estimation |
| 188 | Decomposition Techniques |
| 189 | Empirical Estimation Models |
| 190 | Estimation for Object-Oriented Projects |
| 191 | Specialized Estimation Techniques |
| 192 | The Make/Buy Decision |
| 193 | Summary |
|  | **Project Scheduling** |
| 194 | Basic Concepts |
| 195 | Project Scheduling |
| 196 | Defining a Task Set for the Software Project |
| 197 | Defining a Task Network |
| 198 | Scheduling |
| 199 | Earned Value Analysis |
| 200 | Summary |
|  | **Risk Management** |
| 201 | Reactive versus Proactive Risk Strategies |
| 202 | Software Risks |
| 203 | Risk Identification |
| 204 | Risk Projection |
| 205 | Risk Refinement |
| 206 | Risk Mitigation, Monitoring, and Management |
| 207 | The RMMM Plan |
| 208 | Summary |
|  | **Maintenance and Reengineering** |
| 209 | Software Maintenance |
| 210 | Software Supportability |
| 211 | Reengineering |
| 212 | Business Process Reengineering |
| 213 | Software Reengineering |
| 214 | Reverse Engineering |
| 215 | Restructuring |
| 216 | Forward Engineering |
| 217 | The Economics of Reengineering |
| 218 | Summary |
|  | **Part-5: Advanced Topics** |
|  | **Software Process Improvement** |
| 219 | What is SPI |
| 220 | The SPI Process |
| 221 | The CMMI |
| 222 | The People CMM |
| 223 | Other SPI Frameworks |
| 224 | SPI Return on Investment |
| 225 | SPI Trends |
| 226 | Summary |
|  | **Emerging Trends in Software Engineering** |
| 227 | Technology Evolution |
| 228 | Observing Software Engineering Trends |
| 229 | Identifying ‘Soft Trends’ |
| 230 | Technology Directions |
| 231 | Tools-Related Trends |
| 232 | Summary |
|  | **Concluding Comments** |
| 233 | The Importance of Software – Revisited |
| 234 | People and The Way They Build Systems |
| 235 | New Modes for Representing Information |
| 236 | The Long View |
| 237 | The Software Engineer’s Responsibility |
| 238 | A Final Comment |
|  | **Part-6: Appendix** |
| 239 | An Introduction to UML |
| 240 | Object Oriented Concepts |
|  |  |
|  | **SOFTWARE MAINTANANCE** |

# **Basic Mathematics**

* Pure Mathematics 1, Sophie Goldie
* Pure Mathematics 2 and 3, Sophie Goldie

|  |  |
| --- | --- |
|  | **Pure Mathematics (A-Level)** |
|  | **P1: Algebra** |
| 01 | Background Algebra |
| 02 | Linear Equations |
| 03 | Changing the Subject of a Formula |
| 04 | Quadratic Equations |
| 05 | Solving Quadratic Equations |
| 06 | Equations that cannot be Factorized |
| 07 | The Graphs of Quadratic Function |
| 08 | The Quadratic Formula |
| 09 | Inequalities |
|  | **P2: Algebra** |
| 10 | Operations with Polynomials |
| 11 | Solution of Polynomial Equations |
| 12 | The Modulus Function |
|  | **P3: Further Algebra** |
| 13 | The General Binomial Expansion |
| 14 | Review of Algebraic Functions |
| 15 | Partial Functions |
| 16 | Using Partial Functions with The Binomial Expansion |
|  | **P1: Co-Ordinate Geometry** |
| 17 | Co-Ordinates |
| 18 | Plotting, Sketching and Drawing |
| 19 | The Gradient of a Line |
| 20 | The Distance Between Two Points |
| 21 | The Mid-Point of a Line Joining Two Points |
| 22 | The Equation of a Straight Line |
| 23 | Finding the Equation of a Line |
| 24 | The Intersection of Two Line |
| 25 | Drawing Curves |
| 26 | The Intersection of A Line and A Curve |
|  | **P1: Sequences and Series** |
| 27 | Definitions and Notation |
| 28 | Arithmetic Progressions |
| 29 | Geometric Progressions |
| 30 | Binomial Expansions |
|  | **P1: Functions** |
| 31 | The Language of Functions |
| 32 | Composite Functions |
| 33 | Inverse Functions |
|  | **P1: Differentiation** |
| 34 | The gradient of a Curve |
| 35 | Finding the Gradient of a Curve |
| 36 | Finding the Gradient from First Principles |
| 37 | Differentiating by Using Standard Results |
| 38 | Using Differentiation |
| 39 | Tangents and Normals |
| 40 | Maximum and Minimum Points |
| 41 | Increasing and Decreasing Functions |
| 42 | Points of Inflection |
| 43 | The Second Derivative |
| 44 | Applications |
| 45 | The Chain Rule |
|  | **P2: Differentiation** |
| 46 | The Product Rule |
| 47 | The Quotient Rule |
| 48 | Differentiating Natural Logarithms and Exponentials |
| 49 | Differentiating Trigonometrical Functions |
| 50 | Differentiating Functions Defined Implicitly |
| 51 | Parametric Equations |
| 52 | Parametric Differentiation |
|  | **P3: Differential Equations** |
| 53 | Forming Differential Equations from Rates of Change |
| 54 | Solving Differential Equations |
|  | **P1: Integration** |
| 55 | Reversing Differentiation |
| 56 | Finding the Area Under A Curve |
| 57 | Area as The Limit of A Sum |
| 58 | Areas Below the X Axis |
| 59 | The Area Between Tow Curves |
| 60 | The Area Between A Curve and the Y Axis |
| 61 | The Reverse Chain Rule |
| 62 | Improper Integrals |
| 63 | Finding Volumes by Integration |
|  | **P2: Integration** |
| 64 | Integrals Involving The Exponential Function |
| 65 | Integrals Involving The Natural Logarithm Function |
| 66 | Integrals Involving Trigonometrical Functions |
| 67 | Numerical Integration |
|  | **P3: Further Integration** |
| 68 | Integration by Substitution |
| 69 | Integrals Involving Exponentials and Natural Logarithms |
| 70 | Integrals Involving Trigonometrical Function |
| 71 | The Use of Partial Fractions in Integration |
| 72 | Integration by Parts |
| 73 | General Integration |
|  | **P1: Trigonometry** |
| 74 | Trigonometry Background |
| 75 | Trigonometrical Functions |
| 76 | Trigonometrical Functions for Angles of any Size |
| 77 | The sine and cosine Graphs |
| 78 | The tangent graph |
| 79 | Solving Equations using Graphs of Trigonometrical Functions |
| 80 | Circular Measure |
| 81 | The Length of An ARC of A Circle |
| 82 | The Area of A Sector of A Circle |
| 83 | Other Trigonometrical Functions |
|  | **P2: Trigonometry** |
| 84 | Reciprocal Trigonometrical Functions |
| 85 | Compound-Angle Formulae |
| 86 | Double-Angle Formulae |
| 87 | The Forms rcos, rsin |
| 88 | The General Solutions of Trigonometrical Equations |
|  | **P1: Vectors** |
| 89 | Vectors in Two Dimensions |
| 90 | Vectors in Three Dimensions |
| 91 | Vectors Calculations |
| 92 | The Angle Between Two Vectors |
|  | **P3: Vectors** |
| 93 | The Vector Equation of a Line |
| 94 | The Intersection of Two Lines |
| 95 | The Angle Between Two Lines |
| 96 | The Perpendicular Distance from A Point to a Line |
| 97 | The Vector Equation of a Plane |
| 98 | The Intersection of A Line and A Plane |
| 99 | The Distance of A Point from A Plane |
| 100 | The Angle Between A Line and A Plane |
| 101 | The Intersection of Two Planes |
|  | **P2: Logarithms and Exponentials** |
| 102 | Logarithms |
| 103 | Exponential Functions |
| 104 | Modelling Curves |
| 105 | The Natural Logarithm Functions |
| 106 | The Exponential Function |
|  | **P2: Numerical Solution of Equations** |
| 107 | Interval Estimation – Change-of-Sign Methods |
| 108 | Fixed-Point Iteration |
|  | **P3: Complex Numbers** |
| 109 | The Growth of the Number System |
| 110 | Working with Complex Numbers |
| 111 | Representing Complex Numbers Geometrically |
| 112 | Sets of Points in An Argand Diagram |
| 113 | The Modulus-Argument form of Complex Numbers |
| 114 | Sets of Points Using The Polar Form |
| 115 | Working with Complex Numbers in Polar Form |
| 116 | Complex Exponents |
| 117 | Complex Numbers and Equations |

|  |  |  |
| --- | --- | --- |
| **Each Day: 5 Subjects x 3 Topics = 15 Topics** | | |
| 01 | Programming in C | 104 |
|  | Object-Oriented Programming with C++ | 172 |
| 02 | Fundamental of Database System | 190 |
| 03 | Data Structure and Algorithm | 92 |
| 04 | Software Engineering and Maintenance | 241 |
| 05 | Basic Mathematics | 118 |