NAME OF DEPARTMENT: Computer Applications

Course Name: Bachelor of Science (Hons.) in Computer Science.

Subject Name: Data Structures and File Organization **Subject Code:** TBS 201

1 **Contact Hours:** 48 **L** 3 **T** 1 **P** 0

2 **Examination Duration (Hrs):** **Theory** 0 3 **Practical** 0

3 **Relative Weightage:** **CWE:** 25 **MTE:** 25 **ETE:** 50

4 **Credits:** 0 4

5 **Semester:** ☐ **Autumn** ☑ **Spring** ☐ **Both**

6 **Pre-Requisite:** Basic knowledge of the C programming Concepts

7 **Subject Area:** Programming

8 **Objective:** To familiarize students with the basic knowledge of the data structures which are applied at the various levels of the computer science.

9 **Course Outcome:** A student who successfully fulfills the course requirements will be able to-

- **a.** Have a comprehensive knowledge of the data structures and algorithms on which file structures and data bases are based.
- **b.** Understand the importance of data and be able to identify the data requirements for an application.
- **c.** Have a solid understanding and practical experience of algorithmic design and implementation.
- **d.** Understand the basics of searching and sorting algorithms.
- **e.** Understand the concepts of trees in data structure.

10 Details of the Course:

Unit No.	CONTENT	CONTACT HOURS
1	**Algorithm Basics:** Basic Terminology, types and characteristics, Algorithm Complexity and Time-Space trade-off. **Introduction to Data Structure:** Elementary Data Organization, Data Structure operations, Pointers Review; Static and Dynamic Memory Allocation; **Arrays**: Definition, Representation and Analysis, Single and Multidimensional Arrays, address calculation, application of arrays, Character, Array as Parameters, Ordered list, Sparse Matrices, and Vector.	8
2	**Stacks**: Array Representation and Implementation of stack, Operations and Stacks: Push and POP, Array Representation of Stack, Linked Representation of stack, Operations Associated with Stacks, Application of stack, Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack. **Queues**: Array and linked representation and implementation of queues,	10

	Operations on Queue; Create, Add, Delete, Full and Empty, Circular queue, De-que, and Priority Queue.	
3	**Link List (s):** Representation and implementation of Singly linked lists, Header Linked List, Insertion and deletion to from Linked Lists Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion Algorithms, Doubly linked list, Polynomial representation and addition, Generalized linked list, Garbage Collection and Compaction.	10
4	**Searching**: Linear and Binary searching, Hashing, Hash table, Hash collisions removal techniques. **Sorting**: Bubble sort technique, Selection sort technique, Insertion sort technique, Merge sort technique, quick sort technique. **Recursion**: definition, types of recursion, call stack, iteration – recursion difference.	10
5	**Trees**: Basic terminology, types; **Binary Trees:** Binary Tree, Binary tree representation algebraic Expressions, Complete Binary Tree, Extended Binary Tree, Array and Linked Representation of Binary trees, Traversing Binary trees; **Binary Search Trees:** insertion, traversing, search, deletion operations Threaded binary tree, AVL tree (Height balance tree), **Huffman algorithm:** Huffman tree and Codes; **Multi-way Search trees:** B –Trees, B^+ Tree; **File organizations**: Introduction to file system, Sequential file organization, Index sequential organization, Relative file organization, Multi-key file organization.	10
	TOTAL	**48**

11 Suggested Books:

Sl. NO.	NAME OF AUTHERS/BOOKS/PUBLISHERS	YEAR OF PUBLICATION
1	G.S.Baluja "Data Structure through C", Dhanpat Rai	2007
2	Lipschutz 'Data Structure", TMH	2002
3	Data Structure and Program Design in C-Tanenbaum	1998

Course Name: Object Oriented programming Using C++

Program Name: Bachelor of Science (Hons.) in Computer Science. **Subject Code:** TBS 202

1 Contact Hours: 45 **L** 3 **T** 0 **P** 0

2 Examination Duration (Hrs): **Theory** 0 3 **Practical** 0 0

3 Relative Weightage: **CWE:** 25 **MTE:** 25 **ETE:** 50

4 Credits: 0 3

5 Semester: ☐ **Autumn** ☑ **Spring** ☐ **Both**

6 Pre-Requisite: Basic knowledge of the C programming Concepts

7 Subject Area: Programming

8 Objective: To familiarize students with the Object-Oriented Concepts and its implementation through the C ++ Language.

9 Course Outcome: A student who successfully fulfills the course requirements will be able -

CO1 To describe the differences between procedure-oriented programming and object-oriented programming.
CO2 To implement the basic concepts of the object-oriented programming language like encapsulation, abstraction, inheritance, and polymorphism etc.
CO3 To define the object-oriented approach by implementing various streams, classes, member functions and objects.
CO4 To create objects, array of objects, and pointer to an object of a class.
CO5 Describe how to access private, public and protected members of a class.
CO6 To define the object-oriented approach by implementing various streams, classes, member functions and objects.

10 Details of the Course:

Unit No.	CONTENT	CONTACT HOURS
1	**Introduction to OOPS:** Introduction to object-oriented programming concepts and its applications in real life. Key concepts and uses of OOP, Input and Output in C++. Member functions of iostream class, unformatted console I/O operations. Introduction to all the basic features of OOP like class, object, data abstraction, encapsulation, inheritance, polymorphism, message passing etc.	7
2	**Introduction to C++:** Tokens, keywords, identifiers, variables, operators, expression and control structures, If, If..Else, Switch - Repetitive Statements- for, while, do..while; Variable definition, strict type checking, arrays, strings, pointers etc. Default	8

	arguments, structure, runtime memory management. **Classes & Objects**: Class Specification, Objects, Arrays of objects, Dynamic objects, Pointers to objects, Access specifiers, defining member functions, Data hiding, Functions with class: Friend functions, passing objects as arguments, Returning objects, function overloading, friend class;	
3	**Operators:** Scope resolution, memory management, manipulators. Constructors and Destructors; **Functions in C++ :** parameters passing in functions, values return by functions, function overloading, operator overloading, friend class, static members, Default Arguments, Inline Functions, Using Constructors and its types and Destructors in Derived classes, Implicit Derived class object to base class object conversion.	10
4	**Operator overloading:** Operator overloading, limitations of increment/decrement operators, overloading using friend functions. **Inheritance & Polymorphism** : Inheritance and its types, protected members, protected base class inheritance, Inheriting multiple base classes; Inheritance with Constructor & Destructors, Passing parameters to base class constructors, Granting access, Virtual base classes.	10
5	**Virtual Functions:** Introduction, Type fields and switch statements, Virtual functions, Abstract base classes and concrete classes, Dynamic binding, Virtual destructors, Generic Programming: Templates, Function Template, Class Template Stream I/O in C++.	10
	TOTAL	**45**

11 Suggested Books:

Sl. NO.	**NAME OF AUTHERS/BOOKS/PUBLISHERS**	**YEAR OF PUBLICATION**
1	E. Balagurusamy, Object oriented Programming with C++, 7/ed, TMH.	2017
2	Michael Blaha, James Rumbaugh: Object-Oriented Modeling and design with UML.	2006
3	B. Stroustrup, The C++ Programming Language, Fourth edition, Addison-Wesley.	2013
4	P.M. Chilean: Programming in C++ Merril Pub. 1990.	2008
5	G. Booch, Object Oriented Analysis and Design, Addison-Wesley, 3rd edition.	2007
6	Rebecca Wirfs-Brock, et. al: Designing Object Oriented Software”, PHI.	2005
7	Rumbaugh, J., Object Oriented Modeling and Design, Prentice Hall of India.	2006
8	E.R. Tello: Object Oriented Programming of A.I. Addison Wesley Pub. Co.	2007

Course Name: Bachelor of Science (Hons.) in Computer Science.

Subject Name: Software Engineering **Subject Code:** TBI-203

1 **Contact Hours:** 42 **L** 3 **T** 0 **P** 0

2 **Examination Duration(Hrs):** **Theory** 0 3 **Practical** 0 0

3 **Relative Weightage:** **CWE:** 25 **MTE:** 25 **ETE:** 50

4 **Credits:** 0 3

5 **Semester:** ✓ **Autumn** ☐ **Spring** ☐ **Both**

6 **Pre-Requisite:** Basics of Information Technology

7 **Subject Area:** Software Development and Engineering

8 **Objective:** To familiarize students with the procedures and methods of software development and how to manage and produce efficient & cost-effective software systems.

9 **Course Outcome:** A student who successfully fulfills the course requirements will be able to-

CO1 Understand and apply software engineering principles and development life cycle models in real life projects.
CO2 Develop software requirement specification and design documents for software projects.
CO3 Understand and apply design and coding principles in software projects.
CO4 Create and develop test cases using black box and white testing techniques.
CO5 Evaluate software in terms of size, cost and schedule using project management principles.
CO6 Understand and analyze the importance of quality and reliability in software projects

10 **Details of the Course:**

Unit No.	CONTENT	CONTACT HOURS
1	**Introduction:** Introduction to software engineering, Importance of software, The evolving role of software, Software Characteristics, Software Components, Software Applications, Software Crisis, Software engineering problems. **Software Development Life Cycle Models:** Water Fall Model, Incremental Model, RAD, Prototyping, Spiral Model, comparisons, advantages and disadvantages of models.	8

2	**Software Requirement Engineering:** Requirements elicitation, Problem Analysis, Requirement specifications characteristics, Components of SRS, SRS Document. **Software-Design:** Design principles, problem partitioning, abstraction, top down and bottom up-design, Structured approach, functional versus object oriented approach, design specifications and verification, Monitoring and Control, Cohesiveness, coupling, Fourth generation techniques, Functional independence.	8
3	**Coding:** Top-Down and Bottom –Up programming, Structured Programming, Information hiding, programming style and internal documentation. **Testing:** Testing principles, Levels of testing, functional testing, structural testing, test plane, test case specification, reliability assessment, software testing strategies, Verification & validation, Unit testing, Integration Testing, Alpha & Beta testing, system testing and debugging, Software Maintenance.	8
4	**Software Reliability & Quality Assurance:** Reliability issues, Reliability metrics, Role of matrices and measurement, Reliability growth modeling, Software quality, ISO 9000 certification for software industry, SEI capability maturity model, and comparison between ISO & SEI CMM.	9
5	**Software Project Management:** The Management spectrum- (The people, the product, the process, the project), cost estimation, project scheduling, staffing, software configuration management, quality assurance, project monitoring, risk management, Role of management in software development**.** **CASE (Computer Aided Software Engineering):** CASE and its Scope, CASE support in software life cycle, Documentation, Project Management, internal interface, Reverse Software Engineering, Architecture of CASE environment.	9
	TOTAL	**42**

11 Suggested Books:

Sl. NO.	**NAME OF AUTHERS/BOOKS/PUBLISHERS**	**YEAR OF PUBLICATION**
1	Pressman, Roger S., “Software Engineering: A Practitioner’s Approach Ed. Boston: McGraw Hill, 2001	2004
2	Jalote, Pankaj, “Software Engineering Ed.2”, New Delhi: Narosa 2002	2002
3	Schaum’s Series, “Software Engineering”, TMH	1999
4	Ghezzi, Carlo and Others, “Fundamentals of Software Engineering”, PHI	1998
5	Alexis, Leon and Mathews Leon, “Fundamental of Software Engineering”, Vikas	2001
6	Sommerville, Ian, “Software Engineering”, AWL, 2000 Fairly, “Software Engineering”, New Delhi: TMH	2000

Course Name: Computer Organization and Architecture

Program Name: B.Sc.(CS) **Subject Code:** TBS 204

1 **Contact Hours:** 45 **L** 3 **T** 0 **P** 0

2 **Examination Duration(Hrs):** **Theory** 0 3 **Practical** 0 0

3 **Relative Weightage:** **CWE:** 25 **MTE:** 25 **ETE:** 50

4 **Credits:** 0 3

5 **Semester:** ☐ **Autumn** ✓ **Spring** ☐ **Both**

6 **Pre-Requisite:** Basic Digital Electronics

7 **Subject Area:** Electronics

8 **Objective:** To familiarize students with the organization of the computer.

9 **Course Outcome:** After completion of course a student must be able to

- **CO 1** Understand the behavior and basic working principle of a computer system.
- **CO 2** Exhibit a good understanding of the assembly language programming and basic instructions used.
- **CO 3** Assess the central processing unit, register organization and various types of addressing modes. Understand the importance and benefits of parallel processing.
- **CO 4** Cognize and describe the input output organization and data transfer techniques.
- **CO 5** Appraise the various types of memories used in a computer system. Analyze the importance and functionality of cache and virtual memory organization.
- **CO 6** Review the various aspects of computer organization and summarize the working principles of computer system

10 **Details of the Course:**

Unit No.	CONTENT	CONTACT HOURS
1	Register transfer & Micro-operations, Register transfer language, Bus and memory transfers, Arithmetic logic micro operation, shift micro-operations. Basic computer organization and design: computer instructions. instruction cycle, memory-reference instruction, input-output and interrupts.	9
2	Programming the basic computer: Machine language, Assembly language, The assembler, Program loops, programming arithmetic & logic, subroutines, input-output programming. Micro-programmed control: Control memory, address sequencing, micro-program example and design of control unit.	9

3	Central processing unit: General register organization, stacks organization, addressing modes, data transfer and manipulation, program control, parallel processing, pipelining, arithmetic pipeline.	10
4	Input-output organization: Peripherals devices, input-output interface, asynchronous data transfer, modes of transfer, priority interrupt, direct memory access (DMA), input-output processor, serial communication.	8
5	Classification of memories –RAM organization, Static RAM, Dynamic RAM. ROM organization – PROM, EPROM, EEPROM, EAPROM, Memory Hierarchy, Cache Memory, Mapping, locality of references, Virtual Memory, demand paging, Page fault, Page replacement.	9
	TOTAL	**45**

11 Suggested Books:

Sl. NO.	NAME OF AUTHERS/BOOKS/PUBLISHERS	YEAR OF PUBLICATION
1	Mano,M.M,"ComputerSyatem Architecture",3rd Ed., Pearson Education.	2017
2	Jain R. P. "Digital Electronics"4th Ed. Tata McGraw-Hill.	2009
3	Stallings W."Computer organization",10th Ed. Prentice-Hall.	2016
4	John P.Hayes. "Computer organization",3rd Tata McGraw-Hill.	2017
5	Vravice, Zaky & Hamacher, Computer Organization", 5th Ed. McGraw-Hill.	2011

Program Name: Bachelor of Science (Hons.) in Computer Science.

Course Name: Discrete Mathematical Structures **Subject Code:** TBS 205

1 **Contact Hours:** 45 **L** 3 **T** 0 **P** 0

2 **Examination Duration (Hrs):** **Theory** 0 3 **Practical** 0 0

3 **Relative Weightage:** **CWE:** 25 **MTE:** 25 **ETE:** 50

4 **Credits:** 0 3

5 **Semester:** **Autumn** ☐ **Spring** ✓ **Both** ☐

6 **Pre-Requisite:** Basic knowledge of logic gates and Graphs

7 **Subject Area:** Mathematics

8 **Objective:** To familiarize students with the working of the Graphs used in computer science.

9 **Course Outcome:** A student who successfully fulfills the course requirements will be able to-

a. Understand the theory and techniques of logic, graphs and trees, and algebraic systems.
b. Apply the knowledge and skills obtained to investigate and solve a variety of discrete mathematical problems.
c. Communicate mathematical ideas.
d. Make effective use of appropriate technology.
e. Understand the concept of Paths, Cycles, cut vertex, cut set and bridges.

10 **Details of the Course:**

Unit No.	CONTENT	CONTACT HOURS
1	Matrices: Notation and Definition, Types of Matrices, Algebra of Matrices, Transpose of a Matrix, Solution of linear Equations by Matrix method, Rank of matrix, Eigen values and Eigen vectors, Cayley Hamilton theorem.	10
2	Boolean algebra: Basic operations, Boolean functions, Boolean expression, De-Morgan's theorem, Logic gates, SOP and POS forms, Normal forms, Simplification of Boolean expression, Logic and switching networks, Karnaugh map method for simplification of Boolean expression	10
3	Graph theory: Definition and application of graphs, Konigsberg bridge problem, Simple graph, multi graph and pseudo graph, directed and undirected graphs, degree of a vertex, handshaking theorem, Types of graphs, sub graphs and isomorphic graphs, bipartite graphs, operations of graphs, representation of	10

	graphs.	
4	Paths, Cycles, cut vertex, cut set and bridge, Connectedness in directed and undirected graphs, Connectivity, Eulerian graph, Hamiltonian graph, Dijkstra's algorithm for shortest path, planar graphs, Euler's formula, Graph coloring, Wetch Powell algorithm, Chromatic polynomial, Decomposition theorem.	7
5	TREES: Trees and their properties, Rooted tree, Spanning tree, minimal spanning tree, fundamental circuits, rank and nullity, Kruskal's algorithm, Binary tree.	8
	TOTAL	**45**

11 Suggested Books:

Sl. NO.	**NAME OF AUTHERS/BOOKS/PUBLISHERS**	**YEAR OF PUBLICATION**
1	J. K. Sharma, "Discrete Mathematics", Macmillan	2006
2	Liptschutz, Seymour, "Discrete Mathematics", TMH	2007
3	Trembley, J.P & R. Manohar, "Discrete Mathematical Structure with Application to Computer Science", TMH	2004
4	Kenneth H. Rosen, "Discrete Mathematics and its applications", TMH	2002
5	Doerr Alan & Levasseur Kenneth, "Applied Discrete Structures for Computer Science", Galgotia Pub. Pvt. Ltd	2001
6	Gersting, "Mathematical Structure for Computer Science", WH Freeman & Macmillan	1998
7	Kumar Rajendra, "Theory of Automata: Languages and Computation", PPM	2009
8	Hopcroft J.E, Ullman J.D., "Introduction to Automata theory, Languages and Computation", Narosa Publishing House, New Delhi	2007

Program Name: Bachelor of Science (Hons.) in Computer Science.

Course Name: Environmental Studies **Subject Code:** TEV-111

1 **Contact Hours:** 30 **L** 2 **T** 0 **P** 0

2 **Examination Duration (Hrs):** **Theory** 0 3 **Practical** 0 0

3 **Relative Weightage:** **CWE:** 25 **MTE:** 25 **ETE:** 50

4 **Credits:** 0 2

5 **Semester:** ☐ **Autumn** ☑ **Spring** ☐ **Both**

6 **Pre-Requisite:** Basic knowledge of Environment

7 **Subject Area:** Environmental Studies

8 **Objective:** To familiarize students with the Environmental issues.

9 **Course Outcome:** A student who successfully fulfills the course requirements will be able to-

- **a.** Understand the natural environment and its relationships with human activities.
- **b.** Characterize and analyze human impacts on the environment.
- **c.** Integrate facts, concepts, and methods from multiple disciplines and apply to environmental problems.
- **d.** Understand the Important Environmental Issues, Management and legislation.
- **e.** Project work/Assignment on recent environmental issues and reporting/ Review writing.

10 **Details of the Course:**

Unit No.	**CONTENT**	**CONTACT HOURS**
1	**Ecology and ecosystem:** General Introduction (Scopes and Importance, Components and Segments), Ecosystem (components and structure, energy and nutrient flow, food chain, food web), UNCED (1972), Earth Summit (1992), Rio+20 (2012), UNFCC, CBD.	6
2	**Natural Resources and Biodiversity:** Energy Resources (Renewable/Non-renewable; Traditional/Alternative) and types (Hydel, Solar, Wind, Geothermal). Forest Resources: types and benefits from forest and Sustainable Forest Management, Forest (Conservation) Act, 1980. Water resources: Water resources in Himalayan region, Dams and their impacts, Rain water harvesting and Watershed development, Law and Policy. Biodiversity: Definition and Types, importance, threats and Hotspots, Biodiversity conservation (in situ, ex situ) threatened categories as per IUCN,	6

	Law and Policy. Remote sensing and Geographical Information System (GIS) and applications in environmental management.	
3	**Environmental Pollution:** Air Pollution: Definition, sources, classification and its effects, control strategies and devices, Law and Policy. Water Pollution: Definition, sources, Impacts and toxic effects of some specific pollutants, measurement of DO/BOD/COD, Bio-accumulation and Bio-magnification, Law and Policy. Waste water treatment (Aerobic and anaerobic) and Sewage Treatment Plant. Thermal Pollution and Radioactive pollution and its hazards, Noise pollution.Soil pollution: Definition, sources and solid waste management.	6
4	**Important Environmental Issues, Management and legislation:** Climate change, global warming, smog, ozone layer depletion, acid rain, floods, river blockades, cloud bursting, landslides and earthquakes effects and mitigation. Environmental Impact Assessment (Aims, objectives, constraints in EIA), Environment Assessment Process, EIS, Environment Audit (Introduction and methodology) and Environmental Certification Sustainable Development, Environment and human health. Environmental Management System (EMS), Environmental (Protection) Act 1986, ISO norms	6
5	**Field /Project Work:** Visit and documentation of protected habitats/ Sites/ Research Institutions/ Industries. Project work/Assignment on recent environmental issues and reporting/ Review writing.	6
	TOTAL	**30**

11 Suggested Books:

Sl. NO.	**NAME OF AUTHERS/BOOKS/PUBLISHERS**
1	J. K. Sharma, "Discrete Mathematics", Macmillan
2	Liptschutz, Seymour, "Discrete Mathematics", TMH
3	Trembley, J.P & R. Manohar, "Discrete Mathematical Structure with Application to Computer Science", TMH
4	Kenneth H. Rosen, " Discrete Mathematics and its applications", TMH
5	Doerr Alan & Levasseur Kenneth, "Applied Discrete Structures for Computer Science", Galgotia Pub. Pvt. Ltd
6	Gersting,"Mathematical Structure for Computer Science",WH Freeman & Macmillan
7	Kumar Rajendra, "Theory of Automata: Languages and Computation", PPM