ShriRawatpuraSarkar University, Raipur

ExaminationScheme&Syllabus

For

Bachelor of Technology

In

Computer Science & Engineering

Semester-VI

(Effectivefromthesession: 2022-23)

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Four Years B.TECH Programme

Scheme of Teaching and Examination

B.TECH - Sixth Semester
Computer Science Engineering

Outcome Based Education (OBE) and Choice Based Credit System (CBCS)

(**Effective from the Academic Year 2022-2023**)

S.No.	Course Code	Course Title	Hours / Week			Credits	Maximum Marks			Sem End Exam Duration (Hrs)
			L	**T**	**P**		Continuous Evaluation	Sem End Exam	Total	
1	EBT04601	Distributed Computing	3	1		4	30	70	100	3 Hrs.
2	EBT04602	Compiler Design	3	1		4	30	70	100	3 Hrs.
3	EBT04603	Artificial Intelligence	3	1	-	4	30	70	100	3 Hrs.
4	EBT04604	Android Development & Web Technology	3	1	-	4	30	70	100	3 Hrs.
5	EBT04651	**Elective - I**	3	1	-	4	30	70	100	3 Hrs.
6	EBT04691	Artificial Intelligence Lab	-	-	2	1	15	35	50	3 Hrs.
7	EBT04692	Mini Project	-	-	2	1	15	35	50	3 Hrs.
8	EBT04693	Android Development & Web Technology Lab	-	-	2	1	15	35	50	3 Hrs.
Total Contact hr. per week: 26			**Total Credit: 23**				**Grand Total Marks:**		**650**	

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 6th Semester

Course Title	**Distributed Computing**				
Course Code	EBT04601				
Semester	**6th Semester**				
Course Credit	**L**	**T**	**P**	**T C**	
	3	**1**	**-**	**4**	
Prerequisites	A distributed computer system consists of multiple software components that are on multiple computers, but run as a single system.				
Course Objectives	1. To provide students with contemporary knowledge in distributed systems 2. To equip students with skills to analyze and design distributed applications. 3. To provide master skills to measure the performance of distributed synchronization algorithms				
Course Contents	**UNIT- I** Characterization of Distributed Systems: Issues, Goals, and Types of distributed systems, Distributed System Models, Hardware concepts, Software Concept **,**Middleware: Models of Middleware, Services offered by middleware, Client Server model. **UNIT-II** Layered Protocols, Interprocess communication (IPC): MPI, Remote Procedure Call (RPC), Remote Object Invocation, Remote Method Invocation (RMI) , Message Oriented Communication, Stream Oriented Communication, Group Communication **UNIT-III** Clock Synchronization, Logical Clocks, Election Algorithms, Mutual Exclusion, Distributed Mutual Exclusion-Classification of mutual Exclusion Algorithm, Requirements of Mutual Exclusion Algorithms, Performance measure., Non Token based Algorithms: Lamport Algorithm, Ricart–Agrawala‘s Algorithm, Maekawa‘s Algorithm, Token Based Algorithms: Suzuki-Kasami‘sBroardcast Algorithms, Singhal‘s Heurastic Algorithm, Raymond‘s Tree based Algorithm, Comparative Performance Analysis.				

	UNIT-IV Desirable Features of global Scheduling algorithm, Task assignment approach, Load balancing approach, load sharing approach, Introduction to process management, process migration, Threads,Virtualization, Clients, Servers, Code Migration **UNIT-V** Introduction to replication and consistency, Data-Centric and ClientCentric Consistency Models, Replica Management, Fault Tolerance: Introduction, Process resilience, Reliable client-server and group communication, Recovery
Course Outcomes	1. Demonstrate knowledge of the basic elements and concepts related to distributed system technologies; 2. Illustrate the middleware technologies that support distributed applications such as RPC, RMI and Object based middleware. 3. Analyze the various techniques used for clock synchronization and mutual exclusion 4. Demonstrate the concepts of Resource and Process management and synchronization algorithms 5. Demonstrate the concepts of Consistency and Replication Management
Text Books	1. Andrew S. Tanenbaum and Maarten Van Steen, ―Distributed Systems: Principles and Paradigms, 2nd edition, Pearson Education 2. George Coulouris, Jean Dollimore, Tim Kindberg, , "Distributed Systems: Concepts and Design", 4th Edition, Pearson Education, 2005.
Reference Books	1. A. S. Tanenbaum and M. V. Steen, "Distributed Systems: Principles and Paradigms", Second Edition, Prentice Hall, 2006. 2. M. L. Liu, ―Distributed Computing Principles and Applications‖, Pearson Addison Wesley, 2004.

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 6th Semester

Course Title	**Compiler Design**				
Course Code	EBT04602				
Semester	**6th Semester**				
Course Credit	**L**	**T**	**P**	**T C**	
	3	**1**	**-**	**4**	
Prerequisites	Compiler design principles provide an in-depth view of translation and optimization process.				
Course Objectives	1. To teach concepts of language translation and phases of compiler design 2. To describe the common forms of parsers 3. To inculcate knowledge of parser by parsing LL parser and LR parser 4. To demonstrate intermediate code using technique of syntax directed translation 5. To Illustrate the various optimization techniques for designing various optimizing compilers				

Course Contents	**UNIT – I** INTRODUCTION TO COMPILERS: Definition of compiler, interpreter and its differences, the phases of a compiler, role of lexical analyzer, regular expressions, finite automata, from regular expressions to finite automata, pass and phases of translation, bootstrapping, LEX-lexical analyzer generator. PARSING: Parsing, role of parser, context free grammar, derivations, parse trees, ambiguity, elimination of left recursion, left factoring, eliminating ambiguity from dangling-else grammar, classes of parsing, top down parsing - backtracking, recursive descent parsing, predictive parsers, LL(1) grammars. **UNIT - II** BOTTOM UP PARSING: Definition of bottom up parsing, handles, handle pruning, stack implementation of shift-reduce parsing, conflicts during shift-reduce parsing, LR grammars, LR parsers-simple LR, canonical LR(CLR) and Look Ahead LR (LALR) parsers, error recovery in parsing, parsing ambiguous grammars, YACC-automatic parser generator. **UNIT - III** SYNTAX DIRECTED TRANSLATION: Syntax directed definition, construction of syntax trees, Sattributed and L-attributed definitions, translation schemes, emitting a translation. INTERMEDIATE CODE GENERATION: intermediate forms of source programs– abstract syntax tree, polish notation and three address code, types of three address statements and its implementation, syntax directed translation into three-address code, translation of simple statements, Boolean expressions and flow-of-control statements. **UNIT – IV** TYPE CHECKING: Definition of type checking, type expressions, type systems, static and dynamic checking of types, specification of a simple type checker, equivalence of type expressions, type conversions, overloading of functions and operators. RUN TIME ENVIRONMENTS: Source language issues, Storage organization, storage-allocation strategies, access to non-local names, parameter passing, symbol tables and language facilities for dynamic storage allocation. **UNIT - V** CODE OPTIMIZATION: Organization of code optimizer, basic blocks and flow graphs, optimization of basic blocks, the principal sources of optimization, the directed acyclic graph (DAG) representation of basic block, global data flow analysis. CODE GENERATION: Machine dependent code generation, object code forms, the target machine, a simple code generator, register allocation and assignment, peephole optimization.
Course Outcomes	1. Use compiler construction tools and describes the Functionality of each stage of compilation process 2. Construct Grammars for Natural Languages and find the Syntactical Errors/Semantic errors during the compilations using parsing techniques 3. Analyze different representations of intermediate code. 4. Construct new compiler for new languages. 5. Participate in GATE, PGECET and other competitive examinations
Text Books	Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman (2007), Compilers Principles, Techniques and Tools, 2nd edition, Pearson Education, New Delhi, India.

Reference Books	1. Alfred V. Aho, Jeffrey D. Ullman (2001), Principles of compiler design, Indian student edition, Pearson Education, New Delhi, India. 2. Kenneth C. Louden (1997), Compiler Construction– Principles and Practice, 1st edition, PWS Publishing. 3. K. L. P Mishra, N. Chandrashekaran (2003), Theory of computer science- Automata Languages and computation, 2nd edition, Prentice Hall of India, New Delhi, India. 4. Andrew W. Appel (2004), Modern Compiler Implementation C, Cambridge University Press, UK.

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 6th Semester

Course Title	**Artificial Intelligence**				
Course Code	EBT04603				
Semester	**6th Semester**				
Course Credit	**L**	**T**	**P**	**T C**	
	3	**1**	**-**	**4**	
Prerequisites	Strong knowledge of MathematicsGood command over programming languages. Good Analytical Skills				
Course Objectives	To develop semantic-based and context-aware systems to acquire, organize process, share and use the knowledge embedded in multimedia content. Research will aim to maximize automation of the complete knowledge lifecycle and achieve semantic interoperability between Web resources and services. The field of Robotics is a multi disciplinary as robots are amazingly complex system comprising mechanical, electrical, electronic H/W and S/W and issues germane to all these.				

Course Contents	**Unit- I** AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation **Unit- II** Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A* ,AO* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions. **Unit- III** Knowledge representation issues, predicate logic- logic programming, semantic nets-frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability, Baye's probabilistic interferences and dempstershafer theory. **Unit-IV** First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductivelearning,Decisiontrees,Explanationbasedlearning,Statistical Learning methods ,Reinforcement Learning. **Unit-V** Expert systems:- Introduction, basic concepts, structure of expert systems, the human element in expert systems how expert systems works, problem areas addressed by expert systems, expert systems success factors, types of expert systems, expert systems and the internet interacts web, knowledge engineering, scope of knowledge, difficulties, in knowledge acquisition methods of knowledge acquisition, machine learning, intelligent agents, selecting an appropriate knowledge acquisition method, societal impacts reasoning in artificial intelligence, inference with rules, with frames: model based reasoning, case based reasoning, explanation & meta knowledge inference with uncertainty representing uncertainty.
Course Outcomes	
Text Books	
Reference Books	1. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", SecondEdition, Pearson Education 2. David Poole, Alan Mackworth, Randy Goebel, "Computational Intelligence : a logical approach", Oxford University Press. 3. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problemsolving", Fourth Edition, Pearson Education. 4. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers.

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 6th Semester

<table>
<tr><td>Course Title</td><td colspan="5">Android Development & Web Technology</td></tr>
<tr><td>Course Code</td><td colspan="5">EBT04604</td></tr>
<tr><td>Semester</td><td colspan="5">6th Semester</td></tr>
<tr><td rowspan="2">Course Credit</td><td>L</td><td>T</td><td>P</td><td>T C</td><td></td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td></td></tr>
<tr><td>Prerequisites</td><td colspan="5"></td></tr>
<tr><td>Course Objectives</td><td colspan="5"></td></tr>
<tr><td>Course Contents</td><td colspan="5">UNIT- I
UNIT-II
UNIT- III</td></tr>
<tr><td></td><td colspan="5">UNIT-IV
UNIT-V</td></tr>
<tr><td>Course Outcomes</td><td colspan="5"></td></tr>
<tr><td>Text Books</td><td colspan="5"></td></tr>
<tr><td>Reference Books</td><td colspan="5"></td></tr>
</table>

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 4th Semester

<table>
<tr><td>Course Title</td><td colspan="5">Database Management System</td></tr>
<tr><td>Course Code</td><td colspan="5">----------</td></tr>
<tr><td>Semester</td><td colspan="5">----------</td></tr>
<tr><td rowspan="2">Course Credit</td><td>L</td><td>T</td><td>P</td><td>T C</td><td></td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td></td></tr>
<tr><td>Prerequisites</td><td colspan="5"></td></tr>
<tr><td>Course Objectives</td><td colspan="5">1.To teach the basic database concepts, applications, data models, schemas and instances.
2.To familiarize Entity Relationship model for a database.
3.To demonstrate the use of constraints and relational algebra operations.
4.To describe the basics of SQL and construct queries using SQL.
5.To emphasize the importance of normalization in databases.
6.To demonstrate the basic concepts of transaction processing and concurrency control.</td></tr>
<tr><td>Course Contents</td><td colspan="5">UNIT - I INTRODUCTION: Introduction and applications of DBMS, Purpose of data base, Data, Independence, Database System architecture- Levels, Mappings, Database, users and DBA. DATABASE DESIGN: Database Design Process, ER Diagrams - Entities, Attributes, Relationships, Constraints, keys, extended ER features, Generalization, Specialization, Aggregation, Conceptual design with the E-Rmodel.

UNIT - II THE RELATIONAL MODEL: Introduction to the relational model, Integrity constraints over relations, Enforcing integrity constraints, Querying relational data, Logical database design: E-R to relational, Introduction to views, Destroying/altering tables and views. RELATIONAL ALGEBRA AND CALCULUS: Preliminaries, relational algebra operators, relational calculus - Tuple and domain relational calculus, expressive power of algebra and calculus.

UNIT - III SQL: Basics of SQL, DDL, DML,DCL, structure – creation, alteration, defining constraints – Primary key, foreign key, unique, not null, check, IN operator, Functions - aggregate functions, Built-in functions – numeric, date, string functions, set operations, sub-queries, correlated sub-queries, Use of group by, having, order by, join and its types, Exist, Any, All , view and its types. transaction control commands – Commit, Rollback, Save point, cursors, stored procedures,Triggers.</td></tr>
</table>

	UNIT - IV SCHEMA REFINEMENT AND NORMAL FORMS: Introduction to schema refinement, functional dependencies, reasoning about FDs. Normal forms: 1NF, 2NF, 3NF, BCNF, properties of decompositions, normalization, schema refinement in database design, case studies. UNIT – V TRANSACTIONS MANAGEMENT: Transaction concept, transaction state, implementation of atomicity and durability, concurrent executions, Serializability, recoverability, implementation of isolation, transaction definition in SQL, testing for Serializability. CONCURRENCY CONTROL AND RECOVERY SYSTEM: Concurrency control, lock based protocols, time-stamp based protocols, validation based protocols, multiple granularity. Recovery system - failure classification, storage structure, recovery and atomicity, log- based recovery, shadow paging, buffer management, failure with loss of non-volatile storage, advanced recovery techniques, remote backup systems
Course Outcomes	1. Use the basic concepts of Database Systems in Database design 2. Apply SQL queries to interact with Database 3. Design a Database using ER Modelling 4. Apply normalization on database design to eliminate anomalies 5. Analyze database transactions and can control them by applying ACID properties.
Text Books	1. Raghurama Krishnan, Johannes Gehrke , Database Management Systems, 3rd edition, Tata McGraw Hill, New Delhi,India. 2. ElmasriNavate, Fundamentals of Database Systems, Pearson Education,India.
Reference Books	3. Abraham Silberschatz, Henry F. Korth, S. Sudarshan (2005), Database System Concepts, 5th edition, McGraw-Hill, New Delhi,India. 4. Peter Rob, Carlos Coronel (2009), Database Systems Design, Implementation and Management, 7thedition.

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 4th Semester

<table>
<tr><td>Course Title</td><td colspan="5">Operating System Lab</td></tr>
<tr><td>Course Code</td><td colspan="5">----------</td></tr>
<tr><td>Semester</td><td colspan="5">4th Semester</td></tr>
<tr><td rowspan="2">Course Credit</td><td>L</td><td>T</td><td>P</td><td>T C</td><td></td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td></td></tr>
<tr><td>Prerequisites</td><td colspan="5"></td></tr>
<tr><td>Course Objectives</td><td colspan="5">1. To learn Unix commands and shell programming
2. To implement various CPU Scheduling Algorithms
3. To implement Process Creation and Inter Process Communication.
4. To implement Deadlock Avoidance and Deadlock Detection Algorithms
5. To implement Page Replacement Algorithms
6. To implement File Organization and File Allocation Strategies</td></tr>
<tr><td>Course Contents</td><td colspan="5">1. Write C programs to implement the various CPU Scheduling Algorithms
2. Implementation of Semaphores
3. Implementation of Shared memory and IPC
4. Bankers Algorithm for Deadlock Avoidance
5. Implementation of Deadlock Detection Algorithm
6. Write C program to implement Threading & Synchronization Applications
7. Implementation of the following Memory Allocation Methods for fixed partition
a. First Fit b) Worst Fit c) Best Fit
8. Implementation of Paging Technique of Memory Management
9. Implementation of the following Page Replacement Algorithms
a. FIFO b) LRU c) LFU
10. Implementation of the various File Organization Techniques
11. Implementation of the following File Allocation Strategies</td></tr>
</table>

Course Outcomes	1. Compare the performance of various CPU Scheduling Algorithms 2. Implement Deadlock avoidance and Detection Algorithms 3. Implement Semaphores 4. Create processes and implement IPC 5. Analyze the performance of the various Page Replacement Algorithms 6. Implement File Organization and File Allocation Strategies
Text Books	1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley 2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.
Reference Books	1. Operating Systems – Internals and Design Principles, William Stallings, Fifth Edition–2005, Pearson Education/PHI 2. Operating System - A Design Approach-Crowley, TMH. 3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 4th Semester

Course Title	**Principles of Programming Lab**				
Course Code	**----------**				
Semester	**4th Semester**				
Course Credit	**L**	**T**	**P**	**T C**	
	3	**1**	**-**	**4**	
Prerequisites					
Course Objectives	1. 2. 3. 4.				
Course Contents	Unit- I Unit- II Unit-III Unit-IV Unit- V				
Course Outcomes	1. 2. 3. 4.				
Text Books	1. 2. 3. 4.				
Reference Books	1. 2. 3. 4				

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 4th Semester

<table>
<tr><td>Course Title</td><td colspan="5">Database Management System LabLab</td></tr>
<tr><td>Course Code</td><td colspan="5">----------</td></tr>
<tr><td>Semester</td><td colspan="5">4th Semester</td></tr>
<tr><td rowspan="2">Course Credit</td><td>L</td><td>T</td><td>P</td><td>T C</td><td></td></tr>
<tr><td>3</td><td>1</td><td>-</td><td>4</td><td></td></tr>
<tr><td>Prerequisites</td><td colspan="5"></td></tr>
<tr><td>Course Objectives</td><td colspan="5">1.
2.
3.
4.</td></tr>
<tr><td>Course Contents</td><td colspan="5">1. Design a Database and create required tables. For e.g. Bank, College Database
2. Apply the constraints like Primary Key , Foreign key, NOT NULL to the tables.
3. Write a sql statement for implementing ALTER,UPDATE and DELETE
4. Write the queries to implement the joins
5. Write the query for implementing the following functions: MAX(),MIN(),AVG(),COUNT()
6. Write the query to implement the concept of Intergrity constrains
7. Write the query to create the views
8. Perform the queries for triggers
9. Perform the following operation for demonstrating the insertion , updation and deletion using the referential integrity constraints.
10. Write the query for creating the users and their role</td></tr>
<tr><td>Course Outcomes</td><td colspan="5">1.
2.
3.
4.</td></tr>
<tr><td>Text Books</td><td colspan="5">1.
2.
3.
4.</td></tr>
</table>

Reference Books	1. 2. 3. 4

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 4th Semester

SHRI RAWATPURA SARKAR UNIVERSITY, RAIPUR, CHHATTISGARH

FACULTY OF ENGINEERING

Department of Computer Science & Engineering

B.Tech – CSE 4th Semester