Fifth Semester

Fifth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTES 501-18	Engineering Science	Enterprise Resource Planning	3	0	0	40	60	100	3
BTCS 501-18	Professional Core Courses	Database Management Systems	3	0	0	40	60	100	3
BTCS 502-18	Professional Core Courses	Formal Language & Automata Theory	3	0	0	40	60	100	3
BTCS 503-18	Professional Core Courses	Software Engineering	3	0	0	40	60	100	3
BTCS 504-18	Professional Core Courses	Computer Networks	3	0	0	40	60	100	3
BTCS XXX-18	Professional Elective	Elective-I	3	0	0	40	60	100	3
MC	Mandatory Courses	Constitution of India/ Essence of Indian Traditional Knowledge	2	-	-	100	-	100	S/US
BTCS 505-18	Professional Core Courses	Database Management Systems Lab	0	0	4	30	20	50	2
BTCS 506-18	Professional Core Courses	Software Engineering Lab	0	0	2	30	20	50	1
BTCS 507-18	Professional Core Courses	Computer Networks Lab	0	0	2	30	20	50	1
BTCS XXX-18	Professional Elective	Elective-I Lab	0	0	2	30	20	50	1
	Professional Training	Industrial *Training	-	-	-	60	40	100	S/US
Total			**20**	**0**	**10**	**520**	**480**	**1000**	**23**

* 4-6 weeks industrial training undertaken after 4th semester in summer vacations.

BTES501-18	**Enterprise Resource Planning**	3L:0T:0P	3 Credits

Course Details:

UNIT 1 INTRODUCTION

ERP: An Overview, Enterprise – An Overview, Benefits of ERP, ERP and Related Technologies, Business Process Reengineering (BPR), Data Warehousing, Data Mining, OLAP, SCM [9hrs., CO1]

UNIT II ERP IMPLEMENTATION

ERP Implementation Lifecycle, Implementation Methodology, Hidden Costs, Organizing the Implementation, Vendors, Consultants and Users, Contracts with Vendors, Consultants and Employees, Project Management and Monitoring   [9hrs., CO2]

UNIT III THE BUSINESS MODULES

Business modules in an ERP Package, Finance, Manufacturing, Human Resources, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution   [9hrs., CO3]

UNIT IV THE ERP MARKET

ERP Market Place, SAP AG, Peoplesoft, Baan, JD Edwards, Oracle, QAD, SSA [9hrs., CO4]

UNIT V ERP – PRESENT AND FUTURE

Turbo Charge the ERP System, EIA, ERP and e-Commerce, ERP and Internet, Future Directions

[6hrs., CO1]

TEXT BOOK

1. Alexis Leon, "ERP Demystified", Tata McGraw Hill, New Delhi, 2000

REFERENCES

1. Joseph A Brady, Ellen F Monk, Bret Wagner, "Concepts in Enterprise Resource Planning", Thompson Course Technology, USA, 2001.

2. Vinod Kumar Garg and Venkitakrishnan N K, "Enterprise Resource Planning – Concepts and Practice", PHI, New Delhi, 2003

Course outcomes: The students at the end will be able;

CO1: To know the basics of ERP

CO2: To understand the key implementation issues of ERP

CO3: To know the business modules of ERP

CO4: To be aware of some popular products in the area of ERP

| **Course Code:** BTCS501-18 | **Course Title:** Database Management Systems | **3L:0T:0P** | **3Credits** |

Detailed Contents:

Module 1:   Database system architecture

Data   Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented Data models, integrity constraints, data manipulation operations.

[7hrs] (CO1,2)

Module 2:   Relational query languages

Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. Relational database   design: Domain   and   data   dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design. Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.   **[10hrs] (CO2,4)**

Module 3:

Storage strategies, Indices, B-trees, hashing.   **[3hrs] (CO3)**

Module 4:   Transaction processing

Concurrency   control,   ACID   property,   Serializability   of scheduling,   Locking   and timestamp   based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.   **[6hrs] (CO3)**

Module 5:   Database Security

Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.   **[8hrs] (CO 4,5)**

Module 6:   Advanced Topics

Object oriented and object relational databases, Logical databases, Web databases, Distributed databases.   **[8hrs] (CO 5)**

Course Outcomes:
At the end of study the student shall be able to:

CO1: write relational algebra expressions for a query and optimize the Developed expressions
CO2: design the databases using ER method and normalization.
CO3: construct the SQL queries for Open source and Commercial DBMS-MYSQL, ORACLE, and DB2.
CO4: determine the transaction atomicity, consistency, isolation, and durability.
CO5: Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Text Books:

1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.**:**

Course Code: BTCS502-18	Course Title: Formal Language & Automata Theory	3L:1T:0P	3Credits	42 Hours

Detailed Contents

Module 1:   Introduction
Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages.   **[3hrs] (CO1)**

Module 2:   Regular languages and finite automata:
Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata (NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, minimization of finite automata.   **[8hrs] (CO2)**

Module 3:   Context-free languages and pushdown automata
Context-free grammars (CFG) and languages (CFL), Chomsky and Greibach normal forms, nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic pushdown automata, closure properties of CFLs.   **[8hrs] (CO3)**

Module 4:   Context-sensitive languages
Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG.   **[5hrs] (CO4)**

Module 5:   Turing machines
The basic model for Turing machines (TM), Turing recognizable (recursively enumerable) and Turing-decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.   **[8hrs] (CO 5)**

Module 6:   Undecidability & Intractablity:
Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice s theorem, undecidable problems about languages.
Intractablity: Notion of tractability/feasibility. The classes NP and co-NP, their importance. Polynomial time many-one reduction. Completeness under this reduction. Cook-Levin theorem: NP-completeness of propositional satisfiability, other variants of satisfiability. NP-complete problems from other domains: graphs (clique, vertex cover, independent sets, Hamiltonian cycle), number problem (partition), set cover   **[12hrs] (CO5)**

Course Outcomes: The student will be able to:
CO1: Write a formal notation for strings, languages and machines.

CO2: Design finite automata to accept a set of strings of a language.

CO3: Design context free grammars to generate strings of context free language .

CO4: Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars

CO5: Distinguish between computability and non-computability and Decidability and undecidability.

Text Books:

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.

Course Code: BTCS503-18	Course Title: Software Engineering	3L:1T:0P	3Credits	42 Hours

Detailed Contents:

Module 1:
Evolution and impact of Software engineering, software life cycle models: Waterfall, prototyping, Evolutionary, and Spiral models. Feasibility study, Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

[10hrs] (CO 1)

Module 2:
Basic issues in software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, object modeling using UML, Object-oriented software development, user interface design. Coding standards and Code review techniques.

[8hrs] (CO2)

Module 3:
Fundamentals of testing, White-box, and black-box testing, Test coverage analysis and test case design techniques, mutation testing, Static and dynamic analysis, Software reliability metrics, reliability growth modeling. **[10hrs] (CO 3)**

Module 4:
Software project management, Project planning and control, cost estimation, project scheduling using PERT and GANTT charts, cost-time relations: Rayleigh-Norden results, quality management **[8hrs] (CO4)**

Module 5:
ISO and SEI CMMI, PSP and Six Sigma. Computer aided software engineering, software maintenance, software reuse, Component-based software development.

[6hrs] (CO5)

Text Books:

1. Roger Pressman, "Software Engineering: A Practitioners Approach,(6th Edition), McGraw Hill, 1997.

Course Outcomes:

At the end of the course the student should be able to:

CO 1: Students should be able to identify the need for engineering approach to software development and various processes of requirements analysis for software engineering problems.

CO 2: Analyse various software engineering models and apply methods for design and development of software projects.

CO 3: Work with various techniques, metrics and strategies for Testing software projects.

CO 4: Identify and apply the principles, processes and main knowledge areas for Software Project Management

CO 5: Proficiently apply standards, CASE tools and techniques for engineering software projects

----

Course Code: BTCS 504-18	Course Title: Computer Networks	3L:1T:0P	3Credits	42 Hours

Detailed Contents:

Module 1: Data Communication Components

Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum.

[8hrs] (CO1)

Module 2: Data Link Layer and Medium Access Sub Layer

Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA, CSMA/CDCDMA/CA.   **[10 hrs] (CO2)**

Module 3: Network Layer

Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.   **[8 hrs] (CO3)**

Module 4: Transport Layer

Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.   **[8 hrs] (CO3)**

Module 5: Application Layer

Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography.

[8 hrs] (CO4)

Course Outcomes: The student will be able to:

CO1: Explain the functions of the different layer of the OSI Protocol;

CO2:. Describe the function of each block of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs);

CO3: Develop the network programming for a given problem related TCP/IP protocol; &

CO4: Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.

Text Books:

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw-Hill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Reference Books:

1. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New International Edition.

Course Code: BTCS505-18	CourseTitle: Database management System lab	0L:0T:4P	2Credits

List of Experiments:

Task 1: Introduction to SQL and installation of SQL Server / Oracle.

Task 2: Data Types, Creating Tables, Retrieval of Rows using Select Statement, Conditional Retrieval of Rows, Alter and Drop Statements.

Task 3: Working with Null Values, Matching a Pattern from a Table, Ordering the Result of a Query, Aggregate Functions, Grouping the Result of a Query, Update and Delete Statements.

Task 4: Set Operators, Nested Queries, Joins, Sequences.

Task 5: Views, Indexes, Database Security and Privileges: Grant and Revoke Commands, Commit and Rollback Commands.

Task 6: PL/SQL Architecture, Assignments and Expressions, Writing PL/SQL Code, Referencing Non-SQL parameters.

Task 7: Stored Procedures and Exception Handling.

Task 8: Triggers and Cursor Management in PL/SQL.

Suggested Tools – MySQL, DB2, Oracle, SQL Server 2012, Postgre SQL, SQL lite

Course Outcomes:

CO1: This practical will enable students to retrieve data from relational databases using SQL.

CO2: students will be able to implement generation of tables using datatypes

CO3: Students will be able to design and execute the various data manipulation queries.

CO4: Students will also learn to execute triggers, cursors, stored procedures etc.

Course Code: BTCS506-18	Course Title: Software Engineering Lab	0L:0T:2P	1 Credits

List of Experiments:

Task 1: Study and usage of OpenProj or similar software to draft a project plan

Task 2: Study and usage of OpenProj or similar software to track the progress of a project

Task 3: Preparation of Software Requirement Specification Document, Design Documents and Testing Phase

Task 4: related documents for some problems

Task 5: Preparation of Software Configuration Management and Risk Management related documents

Task 6: Study and usage of any Design phase CASE tool

Task 7: To perform unit testing and integration testing

Task 8: To perform various white box and black box testing techniques

Task 9: Testing of a web site

Suggested Tools - Visual Paradigm, Rational Software Architect. Visio, Argo UML, Rational Application Developer etc. platforms.

Course Code: BTCS507-18	Course Title: Computer Networks Lab	0L:0T:2P	1 Credits

List of Experiments:

Task 1: To study the different types of Network cables and network topologies.

Task 2: Practically implement and test the cross-wired cable and straight through cable using clamping tool and network lab cable tester.

Task 3: Study and familiarization with various network devices.

~~**Task 4:** Familiariz~~ation with Packet Tracer Simulation tool/any other related tool.

Task 5: Study and Implementation of IP Addressing Schemes

Task 6: Creation of Simple Networking topologies using hubs and switches

Task 7: Simulation of web traffic in Packet Tracer

Task 8: Study and implementation of various router configuration commands

Task 9: Creation of Networks using routers.

Task 10: Configuring networks using the concept of subnetting

Task 11: Practical implementation of basic network command and Network configuration commands like ping, ipconfig, netstat, tracert etc. for troubleshooting network related problems.

Task 12: Configuration of networks using static and default routes.

Course Outcomes:

The students will be able to:

CO1: Know about the various networking devices, tools and also understand the implementation of network topologies;

CO2: Create various networking cables and know how to test these cables;

CO3: Create and configure networks in packet trace rtool using various network devices and topologies;

CO4: Understand IP addressing and configure networks using the subnet in;

CO5: Configure routers using various router configuration commands.

Suggested Tools **- NS2/3, Cisco packet tracer, Netsim etc.**.

Course Code: BTCS 510-18   Course Title: Programming in Python   **3L:0T:0P   3 Credits   42 Hours**

Course Outcomes:				

The students should be able to:

CO1: Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.

CO2: Demonstrate proficiency in handling Strings and File Systems.

~~**CO3:** Create, run~~ and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.

CO4: Interpret the concepts of Object-Oriented Programming as used in Python.

CO5: Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

| Course Code: BTCS 510-18 | Course Title: Programming in Python | 3L:0T:0P | 3 Credits | 42 Hours |

Detailed Contents:

Module 1:
Python Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types.

[8hrs] (CO1)

Module 2:
FILES: File Objects, File Built-in Function [open()], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules
Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, *Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, *Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules
Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules.

[10hrs] (CO1,2)

Module 3:
Regular Expressions: Introduction, Special Symbols and Characters, Res and Python Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules.

[8hrs] (CO 2,3)

Module 4:
GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs
WEB Programming: Introduction, Wed Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application Advanced CGI, Web (HTTP) Servers. **[10hrs] (CO 4,6)**

Module 5:
Database Programming: Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules. **[6 hrs] (CO5)**
Text Books:

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

Course Code: BTCS 513-18	Course Title: Programming in Python Lab	0L:0T:2P	1 Credits	2 Hours/week

Prerequisites: Students should install Python.

List of Experiments:

Task 1: Write a program to demonstrate different number data types in Python.

Task 2: Write a program to perform different Arithmetic Operations on numbers in Python.

Task 3: Write a program to create, concatenate and print a string and accessing sub-string from a given string.

Task 4: Write a python script to print the current date in the following format "Sun May 29 02:26:23 IST 2017"

Task 5: Write a program to create, append, and remove lists in python.

Task 6: Write a program to demonstrate working with tuples in python.

Task 7: Write a program to demonstrate working with dictionaries in python.

Task 8: Write a python program to find largest of three numbers.

Task 9: Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula: c/5 = f-32/9]

Task 10: Write a Python program to construct the following pattern, using a nested for loop   *
```
*
* *
* * *
* * * *
* * *
* *
*
*
```

Task 11: Write a Python script that prints prime numbers less than 20.

Task 12: Write a python program to find factorial of a number using Recursion.

Task 13: Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).

Task 14: Write a python program to define a module to find Fibonacci Numbers and import the module to another program.

Task 15: Write a python program to define a module and import a specific function in that module to another program.

Task 16: Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.

Task 17: Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.

Task 18: Write a Python class to convert an integer to a roman numeral.

Task 19: Write a Python class to implement pow(x, n)

Task 20: Write a Python class to reverse a string word by word.