



# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

### SESSION: 2021-22

<b>School: School Of Computer Science &amp; Information Technology</b>								<b>Batch: 2021-2024</b>					
<b>Department: Department of Computer Applications</b>								<b>Year: 1st</b>					
<b>Course: Bachelor of Science (B.SC(CS))</b>								<b>Semester: I</b>					
S N	Cate - gory	Course Code	Course Name	Periods			Cred its	Evaluation Scheme					Subjec t Total Marks
				L	T	P		Theory			Practical		
								ABQ	MSE	ES E	IP	EX P	
1	PCC	BCS-101	Computer Programming	4	0	0	4	15	25	60	-	-	100
2	PCC	BCS-103	Internet and Web Development	4	0	0	4	15	25	60	-	-	100
3	PCC	BCS-105	Computer fundamentals & emerging Technology	3	0	0	3	15	25	60	-	-	100
4	GE	BS-107	Mathematics-I	3	1	0	4	15	25	60	-	-	100
5	AECC	HSS-107	English & Communication Skills	3	0	0	3	15	25	60	-	-	100
6	GE	BS-101	Electricity & Magnetism	3	1	0	4	15	25	60	-	-	100
7	PCC	BCS-151	Computer Programming Lab	0	0	2	1				60	40	100
8	PCC	BCS-153	Internet and Web Development Lab	0	0	2	1				60	40	100
9	PCC	BCS-157	Personal Computer software Lab	0	0	2	1				60	40	100
<b>Total----&gt;</b>				<b>20</b>	<b>2</b>	<b>6</b>	<b>25</b>						

#### **Abbreviations:**

PCC: Programme Core Courses  
 PEC: Programme Elective Courses  
 AECC: Ability Enhancement Compulsory course  
 GE: General English  
 DSE: Discipline Specific Elective Course  
 L: Lecture  
 T: Tutorial  
 P: Practical

ABQ: Assignment Based Quiz  
 MSE: Mid Semester Examination  
 ESE: End Semester Examination  
 IP: Internal Practical  
 EXP: External Practical  
 PROJ: Project



# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

### SESSION: 2021-22

School: School Of Computer Science & Information Technology										Batch: 2021-2024			
Department: Department of Computer Applications										Year: 1st			
Course: Bachelor of Science (B.SC(CS))										Semester: II			
S N	Cate - gory	Course Code	Course Name	Periods			Cred its	Evaluation Scheme					Subjec t Total Marks
				L	T	P		Theory			Practical		
								A B Q	MS E	ES E	IP	EX P	
1	PCC	BCS-102	Data Structures using C	4	0	0	4	15	25	60	-	-	100
2	PCC	BCS-104	Object Oriented Programming using C++	4	0	0	4	15	25	60	-	-	100
3	PCC	BCS-106	Discrete structure	3	0	0	3	15	25	60	-	-	100
4	GE	EC-108	Digital Electronics	4	0	0	4	15	25	60	-	-	100
5	GE	BS-118	Computational Mathematics-II	3	1	0	4	15	25	60	-	-	100
6	GE	CE-108	Environmental Science and Ecology	2	0	0	2	15	25	60	-	-	100
7	PCC	BCS-152	Data Structures Using C Lab	0	0	2	1				60	40	100
8	PCC	BCS-154	Object Oriented Programming using C++ Lab	0	0	2	1				60	40	100
9	PCC	BCS-158	MATLAB	0	0	2	1				60	40	100
<b>Total----&gt;</b>				<b>20</b>	<b>1</b>	<b>6</b>	<b>24</b>						

#### **Abbreviations:**

PCC: Programme Core Courses	ABQ: Assignment Based Quiz
PEC: Programme Elective Courses	MSE: Mid Semester Examination
AECC: Ability Enhancement Compulsory course	ESE: End Semester Examination
GE: General English	IP: Internal Practical
DSE: Discipline Specific Elective Course	EXP: External Practical
L: Lecture	PROJ: Project
T: Tutorial	
P: Practical	



# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

### SESSION: 2022-23

<b>School: School Of Computer Science &amp; Information Technology</b>								<b>Batch: 2021-2024</b>					
<b>Department: Department of Computer Applications</b>								<b>Year: 2nd</b>					
<b>Course: Bachelor of Science (B.SC(CS))</b>								<b>Semester: III</b>					
S N	Cate - gory	Course Code	Course Name	Periods			Cred it s	Evaluation Scheme					Subjec t Total Marks
				L	T	P		Theory			Practical		
								ABQ	MS E	ES E	IP	EX P	
1	PCC	BCS-201	Operating Systems	4	0	0	4	15	25	60	-	-	100
2	PCC	BCS-203	Computer Architecture & Organization	4	0	0	4	15	25	60	-	-	100
3	PCC	BCS-205	Multimedia Technologies	4	0	0	4	15	25	60	-	-	100
4	PCC	BCS-207	Computer Networks	3	0	0	3	15	25	60	-	-	100
5	PCC	BCS-209	Core Java	4	0	0	4	15	25	60	-	-	100
6	PCC	BCS-211	Database Management System	4	0	0	4	15	25	60	-	-	100
7	PCC	BCS-255	Multimedia Technologies Lab	0	0	2	1				60	40	100
8	PCC	BCS-259	Core java Lab	0	0	2	1				60	40	100
9	PCC	BCS-261	Database Management System Lab	0	0	2	1				60	40	100
<b>Total----&gt;</b>				<b>23</b>	<b>0</b>	<b>6</b>	<b>26</b>						

#### **Abbreviations:**

PCC: Programme Core Courses	ABQ: Assignment Based Quiz
PEC: Programme Elective Courses	MSE: Mid Semester Examination
AECC: Ability Enhancement Compulsory course	ESE: End Semester Examination
GE: General English	IP: Internal Practical
DSE: Discipline Specific Elective Course	EXP: External Practical
L: Lecture	PROJ: Project
T: Tutorial	
P: Practical	



# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

### SESSION: 2022-23

<b>School: School Of Computer Science &amp; Information Technology</b>								<b>Batch: 2021-2024</b>					
<b>Department: Department of Computer Applications</b>								<b>Year: 2nd</b>					
<b>Course: Bachelor of Science (B.SC(CS))</b>								<b>Semester: IV</b>					
S N	Cate - gory	Course Code	Course Name	Periods			Cred its	Evaluation Scheme					Subjec t Total Marks
				L	T	P		Theory			Practical		
								AB Q	MS E	ES E	IP	EX P	
1	PCC	BCS-202	Computer Graphics	4	0	0	4	15	25	60	-	-	100
2	PCC	BCS-204	Design and Analysis of Algorithms	4	0	0	4	15	25	60	-	-	100
3	PCC	BCS-206	Linux and Shell Programming	3	0	0	3	15	25	60	-	-	100
4	GE	BS-220	Applied Numerical Techniques	3	0	0	3	15	25	60	-	-	100
5	PCC	BCS-210	IT Management	3	0	0	3	15	25	60	-	-	100
6	PCC	BCS-212	Rapid Application Development	3	1	0	4	15	25	60	-	-	100
7	PCC	BCS-252	Computer Graphics Lab	0	0	2	1				60	40	100
8	PCC	BCS-254	Linux and Shell Programming Lab	0	0	2	1				60	40	100
9	PCC	BCS-256	Rapid Application Development Lab	0	0	2	1				60	40	100
<b>Total----&gt;</b>				<b>20</b>	<b>1</b>	<b>6</b>	<b>24</b>						

#### **Abbreviations:**

PCC: Programme Core Courses	ABQ: Assignment Based Quiz
PEC: Programme Elective Courses	MSE: Mid Semester Examination
AECC: Ability Enhancement Compulsory course	ESE: End Semester Examination
GE: General English	IP: Internal Practical
DSE: Discipline Specific Elective Course	EXP: External Practical
L: Lecture	PROJ: Project
T: Tutorial	
P: Practical	



# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

### SESSION: 2023-24

School: School Of Computer Science & Information Technology										Batch: 2021-2024				
Department: Department of Computer Applications										Year: 3rd				
Course: Bachelor of Science (B.SC(CS))										Semester: V				
S N	Cate - gory	Course Code	Course Name	Periods			Cred its	Evaluation Scheme					Subje t Total Marks	
				L	T	P		Theory			Practical			
								AB Q	MS E	ES E	IP	EX P		
1	PCC	BCS-301	Software Engineering Principles	4	0	0	4	15	25	60	-	-	100	
2	PCC	BCS-303	Programming Using C#	4	0	0	4	15	25	60	-	-	100	
3	PCC	BCS-305	Artificial Intelligence	4	0	0	4	15	25	60	-	-	100	
4	PCC	BCS-307	Introduction to E-commerce	3	0	0	3	15	25	60	-	-	100	
5	DSE		<b>Elective – I</b>	3	0	0	3	15	25	60	-	-	100	
6	PCC	BCS-353	Programming Using C# Lab	0	0	2	1				60	40	100	
7	PCC	BCS-355	Artificial Intelligence Lab	0	0	2	1				60	40	100	
8	PROJ	BCS-371	Minor project	0	0	8	4					100	100	
<b>Total----&gt;</b>				<b>18</b>	<b>0</b>	<b>12</b>	<b>24</b>							

#### Abbreviations:

PCC: Programme Core Courses	ABQ: Assignment Based Quiz
PEC: Programme Elective Courses	MSE: Mid Semester Examination
AECC: Ability Enhancement Compulsory course	ESE: End Semester Examination
GE: General English	IP: Internal Practical
DSE: Discipline Specific Elective Course	EXP: External Practical
L: Lecture	PROJ: Project
T: Tutorial	
P: Practical	

# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

### SESSION: 2023-24

School: School Of Computer Science & Information Technology								Batch: 2021-2024					
Department: Department of Computer Applications								Year: 3rd					
Course: Bachelor of Science (B.SC(CS))								Semester: VI					
S N	Cate - gory	Course Code	Course Name	Periods			Cre dits	Evaluation Scheme					Subjec t Total Marks
				L	T	P		Theory			Practical		
								AB Q	MS E	ES E	IP	EX P	
1	PCC	BCS-302	Software Project Management	3	0	0	3	15	25	60	-	-	100
2	PCC	BCS-304	Neural Network	4	0	0	4	15	25	60	-	-	100
3	PCC	BCS-310	Cryptography & Data Compression	3	0	0	3	15	25	60	-	-	100
4	PCC	BCS-308	Python Programming	3	0	0	3	15	25	60	-	-	100
5	DSE		<b>Elective – II</b>	3	0	0	3	15	25	60	-	-	100
6	PCC	BCS-354	Neural network lab	0	0	2	1				60	40	100
7	PCC	BCS-358	Python Programming Lab	0	0	2	1				60	40	100
8	PROJ	BCS-372	Major Project	0	0	10	5					100	100
<b>Total----&gt;</b>				<b>16</b>	<b>0</b>	<b>14</b>	<b>23</b>						

#### **Abbreviations:**

PCC: Programme Core Courses	ABQ: Assignment Based Quiz
PEC: Programme Elective Courses	MSE: Mid Semester Examination
AECC: Ability Enhancement Compulsory course	ESE: End Semester Examination
GE: General English	IP: Internal Practical
DSE: Discipline Specific Elective Course	EXP: External Practical
L: Lecture	PROJ: Project
T: Tutorial	
P: Practical	

# LINGAYA'S VIDYAPEETH

## SCHEME OF STUDIES

<b>Elective I Courses</b>		
<b>S.No</b>	<b>Course Code</b>	<b>Course Name</b>
1	BCS-309A	Network Security Management
2	BCS-309B	Compiler Design
<b>Elective II Courses</b>		
<b>S.No</b>	<b>Course Code</b>	<b>Course Name</b>
1	BCS-310A	Big Data Analysis
2	BCS-310B	Cloud Computing

**Syllabus**  
**of**  
**BSC**  
**1<sup>st</sup> Year**  
**1<sup>st</sup> Semester**





BCS-101	COMPUTER PROGRAMMING	L-T-P	Cr
		4-0-0	4

## **OBJECTIVE**

To introduce the students the basic of C and Logic behind the implementation of different features of C like different data types , function, array, control statements, pointers, structures, file processing and recursion.

## **COURSE OUTCOMES**

The student after undergoing this course will be able:

**CO1:** To know the different programming languages

**CO2:** To learn the basic concepts of C programming language

**CO3:** To learn the concepts of different control statements

**CO4:** To know about different data types and the ways of handling

**CO5:** To store the data in a file type and how to maintain it

## **UNIT I**

**COMPUTER FUNDAMENTALS AND OVERVIEW OF C PROGRAMMING:** Computer Fundamentals: Algorithm, Flow charts and their symbols. Types of programming languages (Machine Language, Assembly Languages, High level Languages), Introduction to Compiler, Assembler, and Interpreter, Introduction to C, Structure of C program, C character set, Identifier and Keywords, Data types, constants, variables, Declaration, expressions, Types of operators, Input and output functions in C, header files

## **UNIT II**

**CONTROL STATEMENTS:** Control Statements & loops: Sequencing, Selection: if. If-Else, Nesting and switch statements, iteration: for loop, while loop, do while loop, break & continue statements

## **UNIT III**

**ARRAYS AND STRING HANDLING:** Declaring, Referencing and initializing arrays, array subscript, using for loop for sequential access, Using array element as a function argument, multi-dimensional array, String basics string library functions, assignment and substring, concatenation, string comparison.

## **UNIT IV**

**FUNCTION AND POINTERS:** Defining a function, accessing a function, function prototypes, passing arguments to a function, call by value and reference, Types of storage classes, Scope of variable: Global, local, static variables, Recursion. Pointer variables, Declaring & initializing pointers, operations on pointers, pointer expressions, pointers and arrays, pointer and functions, C's dynamic allocation functions.

## **UNIT V**

**STRUCTURE, UNION AND FILE PROCESSING:** Declaration and Initialization of structure, accessing members of a structure, Union, Pre-processor directives, file processing: Introduction, streams and file types, opening and closing a data file, input and output operations, text mode versus binary mode, formatted input output operations with files, structures read and write in files

## **TEXT BOOK**

1. Computer Fundamentals, P.K Sinha, 5th Edition, BPB Publications.
2. Yashwant kanitkar,” Let Us C”, by BPB Publication.

## **REFERENCE BOOKS**

1. Dennis, M. Ritchie and Brian, W. Kernigham, “The C Programming Language, Prentice Hall of India, 2nd Edition, 1988

2. Hanly Jeri R, & E. B. Koffman , “Problem Solving and Program Design in C”, Pearson Publication, 5th Edition, 2008.
3. Byron, C. Gottfried, “Theory and Problem of Programming with C”, Tata McGraw Hill
4. E.Balagurusamy “C – programming” Tata McGray Hill.
5. Schildt, Herbert “The Complete Reference C”, 4th Edition, Tata McGraw Hill, 2004



BCS-103	INTERNET AND WEB DEVELOPMENT	L-T-P	Cr
		4-0-0	4

### **OBJECTIVE**

It aims to provide students will be familiarized with Internet Structure and with the basic protocols which provides knowledge of a proficiency in basic techniques for the development of Web-based applications.

### **Pre-Requisites:**

Knowledge of Web, and basics of Computer and Internet.

### **COURSE OUTCOMES**

The students undergoing this course will be able:

CO1: To know about the web and web hosting

CO2: To create their own website using HTML

CO3: To learn to make the dynamic website using CSS

CO4: To understand the client side programming using Javascript

CO5: To aware about the search engine and its optimization

### **UNIT I**

**INTRODUCTION TO WEB AND HOSTING:** The idea of hypertext and hyper media; how the browser works: MIME types, plug-ins and helper applications; XML, XHTML, XSLT and the W3C, Hosting and Domains: Choosing a domain name, Selecting a hosting company ,Blog hosting services and how they could work for your website; Hosting management tools through the browser such as cPanel.

### **UNIT II**

**HYPertext MARKUP LANGUAGE:** The anatomy of an HTML document; marking up for structure and style: ordered and unordered lists, Structuring content with HTML using natural divisions , Marquee, Anchor Tag, Email Link; embedding images and controlling appearance, table creation: Frames and Nesting, iframes, forms, Semantic elements of HTML5, HTML5 Form elements, Media tags in HTML5, HTML5 Data Storage

### **UNIT III**

**POWER OF CSS:** Introduction to Cascading Style Sheet: Selector, Declaration and declaration block. Types of CSS: Inline, Internal and Internal style specifications within HTML. Types of Selector; Building & Applying Class Selectors; ID Selector using Div Tags; CSS Properties: Table, List, Fonts, Link, Margins, Background Colors.

### **UNIT IV**

**CLIENT SIDE PROGRAMMING:** Introduction to JavaScript syntax: output, Comments, variables, functions, operators, conditions, switch, loop; JavaScript object model: Window, Location and History object model; HTML DOM: Introduction to DOM: methods, event handling, JavaScript Functions, Forms validation and regular expressions.

### **UNIT V**

**SEARCH ENGINE OPTIMIZATION:** What is Search Engine Optimization? Natural vs. paid search and a look at how Google and other search engines work. Maximizing natural search with page titles, meta tags and page content; importance of inbound links in search rankings. Search engine marketing.

### **Textbook:**

Uttam K. Roy, “Web Technology”, Oxford Publication

**References Books:**

1. Musciano Chuck, "HTML & XHTML: The Definitive Guide", Bill Kennedy, 4th Edition, 2000.
2. Holzner Steven, "XHTML Black Book", Paraglyph Press, 2000.
3. Guy W. Lecky-Thompson, "Web Programmin", Cengage Learning, 2008.
4. Kamal Raj, "Internet and Web Technologies", Tata McGraw Hill, 2002

**Web References:**

1. <http://W3schools.com>.
2. <http://www.uniweb.be/>
3. <http://www.sagaciousindia.com/>



<b>BCS-105</b>	<b>COMPUTER FUNDAMENTALS &amp; EMERGING TECHNOLOGY</b>	<b>L-T-P</b>	<b>Credits</b>
		<b>3-0-0</b>	<b>3</b>

## **OBJECTIVE**

To introduce the basic concepts of computers as well as different emerging technologies.

## **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To know the history, different types came into existence via generations of the computer

**CO2:** To understand the different hardware and their usage

**CO3:** To learn the different number systems and their conversions

**CO4:** To learn about the operating system and its need

**CO5:** To know about the different technologies to be emerged in different areas

## **UNIT 1**

**INTRODUCTION TO COMPUTERS:** Basics of Computers What is Computer? Characteristics of Computer Data Processing Cycle (Data -Process -Information) Classification of Computer by Data Processed Analog, Digital and Hybrid Computers History and Generation of Computers First to Fifth Generation Computers Classification of Computers by Processing Capabilities Micro, Mini, Mainframe and Super Computer Simple Model of Computer Input Devices CPU (Central Processing Unit) Arithmetic & Logic Unit Control Unit Internal Memory Output Devices Secondary Storage Devices, Input and output devices

## **UNIT II**

Internal / External parts used with Computer cabinet ,Introduction to Mother board ,Type of Processors Dual Core, Core 2 Duo, i2, i3, etc. ,Memory structure and Types of Memory ,RAM (SRAM, DRAM, SD, DDR, etc.) ,ROM (ROM, PROM, EPROM, EEPROM, etc.) ,Slots ,ISA Slots /PCI Slots / Memory Slots, Sockets ,Cables, Serial Cable / Parallel Cable / USB Cable. ,Ports, USB / Serial / Parallel / PS2, Graphic Cards., Data Storage :Introduction, Type of Magnetic Storage Devices, Floppy Disk / Hard Disk / Magnetic Tape / Magnetic Disks, Storage Mechanism of Magnetic Storage Devices, Tracks / Sectors / Clusters / Cylinders, Reading / Writing Data to and from Storage Devices, Seek Time / Rotational Delay – Latency / Access Time / Response Time, Other Storage Devices, USB – Pen Drive / CD / DVD / Blu-Ray Disk etc.

## **UNIT III**

Numbering System and Codes, Introduction to Binary Codes, Nibble / Bit / Byte / Carry Bit / Parity Bit / Sign Bit, KB / MB / GB / TB / HB / etc., Type of Numbering System, Binary / Octal / Decimal / Hexa-Decimal, Conversion, Binary to Octal, Decimal and Hexa-Decimal, Decimal to Binary, Octal and Hexa-Decimal, Octal to Binary, Decimal and Hexa-Decimal, Hexa-Decimal to Binary, Octal and Decimal, Binary Arithmetic, Addition, Subtraction (1's Compliment and 2's Compliment), Division, Multiplication, Type of Code, ASCII / BCD / EBCDIC / Unicode, Parity Check, Event Parity System / Odd Parity System

## **UNIT IV**

Languages, Operation System and Software Packages, Introduction, Types of Languages (Assembler / Compiler / Interpreter), Machine Level Language, Assembly Level Language, High Level Language (3GL, 4GL, 5GL, etc.), Type of Operation Systems, Batch Operating System, Multi-Processing Operating System., Time Sharing Operating System, Online and Real Time Operating System, Type of Software Packages, Word Processing Package,s, Spread Sheet Packages, Graphical Packages, Database Packages, Presentation Packages, Animation / Video / Sound Packages

## **UNIT V**

Emerging Technologies and Virus, Introduction, Different Communication Methods, GIS / GPS / CDMA / GSM, Communication Devices, Cell Phones / modem / Infrared / Bluetooth / Wi-Fi, Virus, Introduction to Virus and related terms, Origin and History, Types of Virus, Problems and Protection from Virus,. Important Terms and Acronyms, ATM, Backup / Restore, Hard Copy / Soft Copy, Bus / Data Bus, Buffer and Types / Spooling, Cursor / Pointer / Icon, E-Mail / Attachment, CLI / GUI, Compiler and its types, Drive / Directory (Folder) / File / Path, Menu / Pop-up Menu / Toolbar, Shutdown / Reboot, / Restart, Syntax / Wild Card Characters, Optical Fibre (Fibre Optics), Net Meeting, UPS, Printing Speed (CPS, CPM, LPM, DPI, PPM) Peripherals

### **TEXT BOOKS**

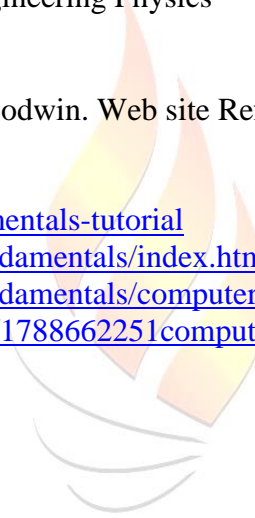
1. Computer Fundamentals By P.K.Sinha
2. Fundamental of IT for BCA By S.Jaiswal Engineering Physics

### **REFERENCE BOOKS**

By V.K.Gaur 4 Teach Yourself Assembler By Goodwin. Web site References : ♣

### **WEB REFERENCES**

1. <https://www.javatpoint.com/computer-fundamentals-tutorial>
2. [https://www.tutorialspoint.com/computer\\_fundamentals/index.htm](https://www.tutorialspoint.com/computer_fundamentals/index.htm)
3. [https://www.tutorialspoint.com/computer\\_fundamentals/computer\\_fundamentals\\_tutorial.pdf](https://www.tutorialspoint.com/computer_fundamentals/computer_fundamentals_tutorial.pdf)
4. [http://www.kvadilabad.org/admin/downloads/1788662251computer\\_fundamentals\\_tutorial.pdf](http://www.kvadilabad.org/admin/downloads/1788662251computer_fundamentals_tutorial.pdf)



BS-107	MATHEMATICS-I	L-T-P	Credits
		3-1-0	4

### **OBJECTIVE**

To introduce the basics concept of mathematics

### **COURSE OUTCOMES**

The students undergoing this course will able:

CO1: To learn about limit, continuity as well as discontinuity

CO2: To state about the matrices and its applications

CO3: To learn about the infinite series

CO4: To know about the differentiation, partial differentiation and its applications

CO5: To aware about the different concepts of integration

### **UNIT I**

**LIMIT & CONTINUITY:** The real line and its geometrical representation;  $\epsilon$ - $\delta$  treatment of limit and continuity; Properties of limit and classification of discontinuities; Properties of continuous functions.

### **UNIT II**

**MATRICES AND ITS APPLICATIONS:** Elementary transformations; inverse of the matrix using elementary transformation; normal form of a matrix; rank of a matrix; solution of simultaneous linear equations; linear dependence and independence of vectors; linear and orthogonal transformations; eigen values, eigen vectors and properties; Cayley-Hamilton theorem and its applications; diagonalization of matrices.

### **UNIT III**

**INFINITE SERIES:** Convergence and divergence; comparison test; D'Alembert's ratio test; Cauchy's root test; Raabe's test; logarithmic test; Gauss test; Cauchy's integral test; Leibnitz's alternate series test; absolutely convergent; conditionally convergent.

### **UNIT IV**

**APPLICATIONS OF DIFFERENTIATION & PARTIAL DIFFERENTIATION:** Asymptotes; Concavity, convexity and points of inflection; Curvature; Extrema; elementary curves, tangent and normal in parametric form; Polar Coordinates. Limits and continuity of functions of two variables; Partial derivatives; Taylor's theorem and Maclaurin's Theorem for function of two variable.

### **UNIT V**

**MULTIPLE INTEGRATIONS:** Double and triple integrals; Change of order in double integrals.

**APPLICATIONS OF INTEGRAION:** length of a curve; Arc length as a parameter; Evoute & Envelope; Volumes and surface areas of solids of revolution.

### **TEXT BOOK:**

Grewal, B.S., "Higher Engineering Mathematics", 41<sup>st</sup> Edition, 2010, Khanna Publishers.

### **REFERENCE BOOKS**

1. Kreyszig, E., "Advance Engineering Mathematics", 10<sup>th</sup> Edition, 2011, Wiley India Publishers, New Delhi
2. Weir, M. D., Hass, J. and Giordano, F. R., "Thomas Calculus", 11<sup>th</sup> Edition, 2012, Pearson Education.

3. Jain, R.K. and Iyengar, S.R.K., “ Advance Engineering Mathematics” ,3<sup>rd</sup> Edition,2002, Narosa Publishing House New Delhi.
4. H. Anton, I. Bivens and S. Davis, *Calculus*, 7th Ed., John Wiley and Sons (Asia) P. Ltd., Singapore, 2002.
5. G.B. Thomas and R.L. Finney, *Calculus*, 9th Ed., Pearson Education, Delhi, 2005.





<b>HSS-107</b>	<b>ENGLISH &amp; COMMUNICATION SKILLS</b>	<b>L-T-P</b>	<b>Cr.</b>
		<b>3-0-0</b>	<b>3</b>

## **OBJECTIVE**

To make students understand the concepts related to language development communication skills.

## **COURSE OUTCOMES**

The students undergoing the course will be able:

**CO1:** To know the basic structure of speech

**CO2:** To learn about oral communication and role of speech organs in it

**CO3:** To get knowledge about the writing skills

**CO4:** To be skillful in writing different applications as well as letters

**CO5:** To know all about comprehension

## **UNIT I**

**REMEDIAL ENGLISH:** Parts of Speech; Tenses and their application; Verbs and their various forms.

## **UNIT II**

**ORAL COMMUNICATION:** Introduction to oral communication; Importance of pronunciation; Phonetics; Importance of phonetics; Basic introduction to speech organs, articulation and phonetic symbols.

## **UNIT III**

**WRITING SKILLS:** Introduction to various types of writings including general writing, technical writing, picture composition, Slogan making; Movie review etc.

## **UNIT IV**

**LETTER WRITING:** Types of letter writing; Structure & layout; Leave application; Letter of enquiry & response with respect to educational & official matters; Informal letter expressing or discussing social or educational issues.

## **UNIT V**

**COMPREHENSION:** Listening comprehension & reading comprehension; Listening to recorded speeches, TV news and other audio materials to test listening comprehension; comprehension of unseen passages through reading.

## **TEXT BOOK:**

Pal Rajendra, Korlaha, Hi, J.S., “Essentials of Business Communication”, Sultan Chand & Sons

## **REFERENCE BOOKS**

1. Rutherford, Andrea, J., “Basic Communication Skills for Technology”, Pearson Education Asia.
2. Prasad, V., “Advanced Communication Skills”, Atma Ram Publications, New Delhi.
3. Madhukar, R., K., “Business Communication”, Vikas Publishing House Pvt. Ltd.

BCS-151	COMPUTER PROGRAMMING LAB	L-T-P	Cr.
		0-0-2	1

## OBJECTIVE

To implement different aspects of C Language using different control statements and loops as well as different storage structures like arrays, strings and files.

## COURSE OUTCOMES

The student after undergoing this course will be able:

**CO1:** To implement the different control statements like sequential, conditional & loops

**CO2:** To learn the basic concepts of C programming language

**CO3:** To learn the concepts of different control statements

**CO4:** To know about different data types and the ways of handling

**CO5:** To store the data in a file type and how to maintain it

## List of Experiments

### SEQUENTIAL CONTROL STATEMENTS

1. Write a program to Print HELLO
2. Write a program to add two numbers
3. Write a program to calculate simple interest
4. Write a program to calculate average of three numbers
5. Write a program to swap two numbers
6. Write a program to illustrate mixed data types
7. Write a program to calculate area and circumference of circle
8. Write a program to evaluate a polynomial expression
9. Write a program to add digits of a four digit number
10. Write a program to check whether the person is eligible for voting or not

### CONDITIONAL CONTROL STATEMENTS

11. Write a program to find greatest of two numbers
12. Write a program to find out which type of triangle it is
13. Write a program to find out greatest of three numbers
14. Write a program to evaluate performance of the student
15. Write a program to make a basic calculator

### LOOP CONTROL STATEMENTS

16. Write a program to print fibonacci upto the given limit
17. Write a program to find the sum of digits of a number
18. Write a program to find factorial of a number
19. Write a program to print table of any number

### ARRAYS AND STRINGS

20. Write a program to enter the elements in a one dimensional array
21. Write a program to find the sum and average of five numbers
22. Write a program to sort the array elements
23. Write a program to enter the marks of 50 students and calculate the average
24. Write a program to add 2 matrix
25. Write a program to multiply 2 matrices
26. Write a program to calculate the length of string
27. Write a program to concatenate 2 strings
28. Write a program to reverse the string
29. Write a program to count the numbers of characters in a string
30. Write a program that converts lower case characters to upper case
31. Write a program without using predefined functions to check whether the string is palindrome or not

### FUNCTIONS

32. Write a program using function to find the largest of three numbers

33. Write a program using function to swap two numbers using call by value
34. Write a program using function to swap two numbers using call by reference
35. Write a program using function to sum the digits of a number
36. Write a program to calculate factorial of a number using recursive function
37. Write a program to print first n fibonacci using recursive function

### **POINTERS**

38. Write a program to illustrate the concept of chain of pointers
39. Write a program to calculate the area and perimeter of circle using pointers
40. Write a program to find largest of three numbers

### **STRUCTURES**

41. Write a program to read an employee record using structure and print it
42. Write a program to prepare salary chart of employee using array of structures



BCS-153	INTERNET AND WEB DEVELOPMENT LAB	L-T-P	Cr
		0-0-2	1

**OBJECTIVE:**

It aims to make students to make them skillful in creating and handling web based applications like websites etc.

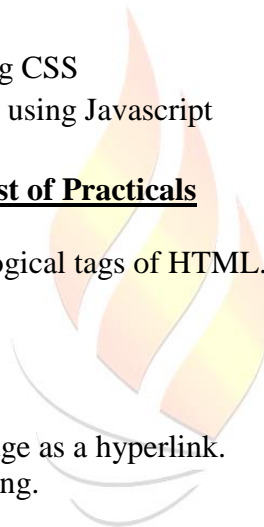
**COURSE OUTCOMES**

The students undergoing this course will be able:

- CO1:** To know and use different HTML tags
- CO2:** To create different lists in a webpage or website
- CO3:** To create their own website
- CO4:** To learn to make the dynamic website using CSS
- CO5:** To understand the client side programming using Javascript

**List of Practicals**

1. Design a web page using: Physical and Logical tags of HTML.
2. Design a web page using
  - 2.2 Ordered List
  - 2.3 Unordered Lists
  - 2.4 Nested Lists
3. Design a web page to show the use of image as a hyperlink.
4. Design a web-page using frames and linking.
5. Code to create a bookmark.
6. Design a web-page showing the use of forms using HTML 4.01 and HTML 5 Tags.
7. Design a page using basic tags of HTML 5.0.
8. Design a web-page using style sheets (External, Internal and Inline)
9. Write a Program to print if the no is even or odd using JavaScript.
10. Input a number and find the difference of the sum of factors and non-factors.
11. WAP in JavaScript to print the pattern
  - 12345
  - 1234
  - 123
  - 12
  - 1
12. WAP to Accept an Array of 10 numbers and display the sum of elements.
13. WAP to find greatest of all elements of an array.
14. Design a web-page to show different validation checking using Java Script.
15. WAP in PHP code for calculating S.I.
16. WAP to Calculate factorial of a number.
17. WAP to print the table of 10.
18. WAP to print the sum of diagonal elements.
19. WAP to enter 5



<b>BCS-157</b>	<b>PERSONAL COMPUTER SOFTWARE LAB</b>	<b>L-T-P</b>	<b>Cr</b>
		<b>0-0-2</b>	<b>1</b>

### **OBJECTIVE**

To make the students computer savvy via introducing different basic applications like MS-office

### **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To learn all windows based commands

**CO2:** To expertise in making documents using MS- Word

**CO3:** To handle in handling database by creating spreadsheets using MS-Excel


**CO4:** To manage the data via sorting, filtering etc.

**CO5:** To be expert in presentations using MS-Powerpoint

### **LIST OF EXPERIMENTS/EXERCISES**

1. All commands specified using windows
2. Introduction to MS-Word: introduction to word processing, its features.
3. MS Word: formatting documents, paragraph formatting, indents.
4. Page formatting, header and footer, bullets and numbering.
5. MS-Word: tabs, tables, formatting the tables, finding and replacing text, mail merging etc.
6. Introduction to MS-Excel, introduction to electronic spreadsheets, feature of MS-Excel
7. Entering data, entering series, editing data, cell referencing, ranges, formulae, functions, auto sum, copying formula
8. MS Excel: formatting data, creating charts, creating database, sorting data, filtering etc.
9. Introduction to MS PowerPoint, PowerPoint, features of MS-PowerPoint clipping, slide animation, slide shows, formatting etc.
10. MS-PowerPoint presentation (10-15 slides) on (i) evolution of computers (ii) search engines

**Syllabus**  
**of**  
**BSC**  
**1<sup>st</sup> Year**  
**2<sup>nd</sup> Semester**



BCS-102	DATA STRUCTURES USING C	L-T-P	Cr
		4-0-0	4

### **OBJECTIVE**

To relay the theoretical and practical fundamental knowledge of most commonly used Data Structures.

### **PRE-REQUISITES**

Knowledge of basic computer programming

### **COURSE OUTCOMES**

**CO1:** Understand the concept of Dynamic memory management, data types, algorithms, Big O notation.

**CO2:** Understand basic data structures such as arrays, linked lists, stacks and queues.

**CO3:** Describe the hash function and concepts of collision and its resolution methods

**CO4:** Solve problem involving graphs, trees and heaps

**CO5:** Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data

### **UNIT I**

**INTRODUCTION TO DATA STRUCTURES AND RUNNING TIME:** Definition of data structures and abstract data types; linear vs. non-linear data types; primitive vs. non-primitive data types; Static and Dynamic implementations; Arrays: 1,2-dimensional arrays, search, insertion, deletion in 1-D. Time Complexity; Big – Oh - notation; Running Times; Best Case; Worst Case; Average Case; Introduction to Recursion.

### **UNIT II**

**STACKS AND QUEUES:** The Stacks: Definition; Array based implementation of stacks; Examples: Infix; postfix; prefix representation; Conversions; definition of Queues; Array based implementation of Queues. Circular queue.

### **UNIT III**

**LINKED LISTS:** Lists; Types of Linked list, Operations on Linked list; Circular Linked list; Linked List based implementation of stacks; Linked List implementation of Queues; Applications.

### **UNIT IV**

**TREES AND GRAPHS:** Definition of trees and Binary trees; Properties of Binary trees and Implementation; Binary Traversal pre-order; post order; In- order traversal; Binary Search Trees. Definition of Undirected and Directed Graphs; The Array based implementation of graphs; Adjacency matrix; path matrix implementation; The Linked List representation of graphs; Graph Traversal – Breadth first Traversal; Depth first Traversal.

### **UNIT V**

**SORTING AND SEARCHING ALGORITHMS:** Introduction; Sorting by exchange; selection; insertions; bubble sort; Merge sort; Quick sort; Searching Algorithms: Straight Sequential Search; Binary Search (recursive & non-recursive Algorithms).

## **TEXT BOOK**

Aho, A.V, Hopcroft, J. E., Ullman, T. D., “Data Structures and Algorithms”, Original Edition, Addison-Wesley, Low Priced Edition, 1999

## **REFERENCE BOOKS**

1. M. Tenenbaum, Langsam, Moshe J. Augentem, “Data Structures using C”, Prentice Hall of India
2. Ellis, Horowitz, SartajSahni, “Fundamentals of Data Structures”, Addison-Wesley Pub, 1983
3. Mark Allen Weiss, “Data Structures and Algorithms Analysis in C”, Pearson Education, 2000





<b>BCS-104</b>	<b>OBJECT ORIENTED PROGRAMMING USING C++</b>	<b>L-T-P</b>	<b>Cr</b>
		<b>4-0-0</b>	<b>4</b>

### OBJECTIVE

To build programming logic and thereby developing skills in problem solving using C++ programming language; Introduce the concept of object orientation and on how to handle data in different forms; Emphasize the concepts and constructs rather than on language features.

### COURSE OUTCOMES

**CO1:** Describe the procedural and object-oriented paradigm with concepts of streams, classes, functions, data and objects.

**CO2:** Understand dynamic memory management techniques using pointers, constructors, destructors, etc

**CO3:** Describe the concept of function overloading, operator overloading, virtual functions and polymorphism.

**CO4:** Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming.

**CO5:** Demonstrate the use of various OOPs concepts with the help of programs.

### UNIT I

**OBJECT ORIENTED CONCEPTS:** Introduction to objects and object oriented programming; Difference between procedural oriented and objected oriented paradigm ,features of objected oriented programming: encapsulation (information hiding); Abstraction, Inheritance, Polymorphism, dynamic binding, access modifiers: introduction to C++, simple C++ programs, function overloading, Inline function, reference type, default arguments, concept of namespace

### UNIT II

**CLASSES AND DATA ABSTRACTION:** Introduction; structure definitions; accessing members of structures; class scope and accessing class members; controlling access to a class; method; or variable (public; protected; private; separating interface from implementation; controlling access function and utility functions; initializing class objects: constructors; default, parameterized and copy constructor; using destructors; classes : const(constant) object and const member functions; object as member of classes; friend function and friend classes; using this pointer; dynamic memory allocation with new and delete; static class members;

### UNIT III

**OPERATOR OVERLOADING& TEMPLATES:** Introduction; fundamentals of operator overloading; restrictions on operators overloading; operator functions as class members vs. as friend functions; overloading; <<; >> overloading unary operators; overloading binary operators. Concept of Templates & generic programming, Function Templates, class templates

### UNIT IV

**INHERITANCE; VIRTUAL FUNCTIONS AND POLYMORPHISM:** Introduction; inheritance: base classes and derived classes; protected members; single, multiple, multilevel, hierarchical, inheritance; overriding base-class members in a derived class; public; protected and private inheritance; using constructors and destructors in derived classes. casting base-class pointers to derived-class pointers;, virtual function & dynamic binding

## **UNIT V**

**FILES, I/O STREAMS& EXCEPTION HANDLING:** Files and streams; stream input/output classes and objects file modes, opening a file, creating & writing to a sequential access file; reading data from a sequential access file; updating sequential access files; unformatted I/O (with read and write);,reading & writing objects, file pointers and related functions, stream manipulators. Basics of C++ exception handling: try; throw; catch; throwing an exception; catching an exception.

## **TEXT BOOK**

Balagurusamy, E., “Object Oriented Programming with C++”, Prentice Hall of India, 2008.

## **REFERENCE BOOKS**

1. Kamthane, “Object Oriented Programming with ANSI and Turbo C++”, Pearson Education
2. Lafore, Robert, “Object Oriented Programming in Turbo C++”, The WAITE Group Press, 1994
3. Schildt, Herbert “C++: The Complete Reference”, Tata McGraw Hill, 3rd Ed, 2008
4. Bhave, “Object Oriented Programming with C++”, Pearson Education

BCS-106	DISCRETE STRUCTURE	L T P	Cr
		3 0 0	3

## OBJECTIVE

To lay mathematical foundation for the fundamentals of various computational structures such as Boolean algebra, propositional logic, graph and trees.

**PRE-REQUISITES:** Knowledge of Data Structure

## COURSE OUTCOMES

**CO1:** Perform operations on various discrete structures such as sets, functions, relations, and sequences.

**CO2:** Ability to solve problems using Counting techniques, Permutation and Combination, Recursion and generating functions.

**CO3:** Apply algorithms and use of graphs and trees as tools to visualize and simplify Problems.

**CO4:** Apply algorithms and use of graphs and trees as tools to visualize and simplify Problems.

**CO5:** Understand the various properties of algebraic systems like Rings, Monoids and Groups.

## UNIT I

**SET THEORY:** Introduction to set theory; set operations; algebra of sets: duality, finite and Infinite sets, classes of sets, power sets, multi sets, Cartesian product, representation of relations, Types of relation, equivalence relations and partitions, partial ordering relations and lattices; Function and its types, composition of function and relations; cardinality and inverse relations

## UNIT II

**PROPOSITIONAL CALCULUS:** Basic operations: AND ( $\wedge$ ), OR ( $\vee$ ), NOT ( $\sim$ ), truth value of a Compound statement, propositions, tautologies, contradictions.

## UNIT III

**TECHNIQUES OF COUNTING:** Permutations with and without repetition, combination.

## UNIT IV

**ALGEBRIC STRUCTURES:** Definition and examples of a monoid, semi group, groups and Rings; homomorphism, isomorphism and auto Orphism; subgroups and normal subgroups; cyclic groups, integral domain and fields; co-sets; Lagrange's theorem

## UNIT V

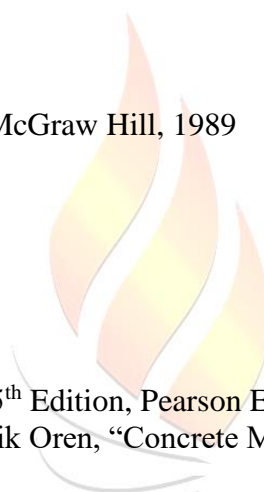
**GRAPHS:** Introduction to graphs, directed and undirected graphs; homomorphic and isomorphic Graphs; sub graphs; cut points and bridges; multigraph and weighted graph; paths and circuits, shortest path in weighted graphs; Eulerian path and circuits, Hamilton paths and circuits; planar Graphs; Euler's formula, Spanning tree

## **TEXT BOOK**

Liu C. L., "Elements of Discrete Mathematics", McGraw Hill, 1989

## **REFERENCE BOOKS**

1. Johnson Bough R., "Discrete Mathematics", 5<sup>th</sup> Edition, Pearson Education, 2001
2. Graham Ronald, Knuth Donald E. and Patashik Oren, "Concrete Mathematics: A Foundation for Computer Science", Addison-Wesley, 1989
3. Gersting Judith L., "Mathematical Structures for Computer Science", Computer Science Press, 1993
4. Chtewynd A. and Diggle P., "Discrete Mathematics", Modular Mathematics Series, Edward Arnold, London, 1995
5. Lipshutz S., "Schaums Outline series: Theory and problems of Probability", McGraw Hill Singapore, 1986.
6. Kolman B. and Busby R. C., "Discrete Mathematical Structures", Prentice Hall of India, 1996
7. Trembley and Manohar, "Discrete Mathematical Structures with Applications to Computers" McGraw Hill, 1995



EC-108	DIGITAL ELECTRONICS	L T P	CR
		4 0 0	4

### **OBJECTIVE**

Modern world deals with digital conditioning of various signals. Digitally manipulating signals or using digital circuits have a lot of advantages in terms of accuracy etc. This subject introduces concept of basic digital electronics: gates; combinational and sequential circuits and their designing.

### **COURSE OUTCOMES**

The students undergoing this course will be able to:

**CO1:** Have a thorough understanding of the fundamental concepts and techniques used in digital electronics.

**CO2:** To understand and examine the structure of various number systems and its application in digital design.

**CO3:** The ability to understand, analyze and design various combinational and sequential circuits.

**CO4:** Ability to identify basic requirements for a design application and propose a cost effective solution.

**CO5:** The ability to identify and prevent various hazards and timing problems in a digital design.

### **UNIT I**

**INTRODUCTION OF GATES, COMBINATIONAL DESIGN BY USING GATES AND SIMPLIFICATION:** Digital signal; logic gates: AND; OR; NOT; NAND; NOR; EX-OR; EX-NOR; Boolean algebra. Review of Number systems. Binary codes: BCD; Excess- 3; Gray; EBCDIC; ASCII; Error detection and correction codes; Design using gates; Karnaugh map and Quine Mcluskey methods of simplification.

### **UNIT II**

**COMBINATIONAL DESIGN USING MSI DEVICES:** Multiplexers and Demultiplexers and their use as logic elements; Decoders; Adders/Subtractors; BCD arithmetic circuits; Encoders; Decoders/Drivers for display devices.

### **UNIT III**

**SEQUENTIAL CIRCUITS:** Flip Flops : S-R; J-K; T; D; master-slave; edge triggered; shift registers; sequence generators; Counters; Asynchronous and Synchronous Ring counters and Johnson Counter; Design of Synchronous and Asynchronous sequential circuits.

### **UNIT IV**

**DIGITAL LOGIC FAMILIES:** Bipolar logic families: RTL; DTL; DCTL; HTL; TTL; ECL; MOS; and CMOS logic families. Tristate logic; Interfacing of CMOS and TTL families.

### **UNIT V**

**A/D AND D/A CONVERTERS & PLD:** Sample and hold circuit; weighted resistor and R<sup>-2</sup> R ladder D/A Converters; specifications for D/A converters. A/D converters: successive approximation; counting type; ROM; PLA; PAL; FPGA and CPLDs.

### **TEXT BOOK**

Jain, R.P., "Modern Digital Electronics", 4th Ed.; Tata McGraw Hill, 2003

## **REFERENCE BOOKS**

1. Taub and Schilling, "Digital Integrated Electronics" Tata McGraw Hill,1997
2. Malvino and Leach; "Digital Principles and Applications", 6th Edition, Tata McGraw Hill, 2006
3. Mano, Morris, "Digital Design", 3rd Edition, Prentice Hall of India,1994
4. Gupta and Singhal, "Digital Electronics", 2nd Edition, Dhanpat Rai and Sons, 2000.
5. Wakerly, John F, "Digital Design Principles and Practices", 4th Edition, Prentice Hall of India,2005



BS-118	COMPUTATIONAL MATHEMATICS-II	L T P	CR
		3-1-0	4

### UNIT I

**MEASURES OF CENTRAL TENDENCY:** Concept and properties of mathematical averages including arithmetic mean, geometric mean and harmonic mean, Mode and Median (and other partition values - quartiles, deciles, and percentiles) with graphic presentation.

### UNIT II

**MEASURES OF DISPERSION:** Range, Quartile deviation, mean deviation, standard deviation, and their coefficients; Properties of standard deviation/variance, Moments: Calculation and significance;

### UNIT III

**THEORY AND APPROACHES OF PROBABILITY:** Theory and approaches of probability, Probability Theorems: Addition and Multiplication (Proof not required). Conditional probability and Bayes' Theorem (Proof not required), Binomial distribution: Probability distribution function, Poisson distribution: Probability function (including Poisson approximation to binomial distribution)

### UNIT IV

**LINEAR PROGRAMMING PROBLEM:** Introduction to Linear Programming Problem, Some definitions, mathematical formulation Linear Programming Problem, Graphical method of solving of Linear Programming Problem. Corner Point method.

### UNIT V

**GAME THEORY:** Two-person-zero sum games, Games of pure strategies and games of mixed strategies, Rule of dominance, Graphic solutions to games, Applications in computer science.

### **RECOMMENDED BOOKS**

1. G.M. Clarke and D Cooke, A Basic Course in Statistics, Arnold, (2004).
2. W. Filler, An introduction to Probability theory and its applications, John Wiley, (1968)
3. A.M. Goon, M.K. Das and B. Dasgupta, Fundamentals of Statistics, World Press, (1997)
4. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, S. Chand and sons, 2002.
5. E.N. Barron, Game Theory: An Introduction, Wiley Student Addition, 2009
6. S. Tijs, INTroduction to Game Theory, Hindustan Book Agency.

CE-108	ENVIRONMENTAL SCIENCE & ECOLOGY	L T P	CR
		2-0-0	2

## OBJECTIVES

- The aim of the course is to make everyone aware of environment issues like continuing problems of pollution, loss of forest, solid waste disposal and degradation of environment.
- Issues like economic productivity and national security, global warming, the depletion of ozone layer and loss of biodiversity are other serious concerns before the mankind.

## COURSE OUTCOMES

**CO1:** Conceptualize the processes and various factors involved in the formation of environment.

**CO2:** Recognize the importance of environment and the sustainable of natural resources.

**CO3:** Analyze interaction between social and environmental processes.

**CO4:** Use scientific reasoning to identify and understand environment problems and evaluate potential solutions.

**CO5:** Visualize the impacts of human activities on environment and role of society in these impacts.

## UNIT I

**THE MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES:** Definition, scope and importance, Need for public awareness, Environment Impact Assessment.

## UNIT II

**NATURAL RESOURCES- RENEWABLE AND NON-RENEWABLE RESOURCES:**

### **A. Natural resources and associated problems.**

- Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forests and tribal people.
- Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems.
- Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.
- Food resources: World food problems, changes caused by agriculture and
- overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water
- logging, salinity, case studies.
- Energy resources: Growing energy needs, renewable and non-renewable energy
- sources, use of alternate energy sources, case studies.
- Land resources: Land as a resource, land degradation, man induced landslides,
- soil erosion and desertification.

### **B. Role of an individual in conservation of natural resources.**

### **C. Equitable use of resources for sustainable lifestyles.**

## UNIT III: ECOSYSTEMS



- A. Concept of an ecosystem
- B. Structure and function of an ecosystem
- C. Producers, consumers and decomposers
- D. Energy flow in the ecosystem
- E. Ecological succession
- F. Food chains, food webs and ecological pyramids
- G. Introduction, types, characteristic features, structure and function of the following ecosystem:
  - Forest ecosystem
  - Grassland ecosystem
  - Desert ecosystem
  - Aquatic ecosystems (ponds, streams, lakes, rivers, ocean estuaries).

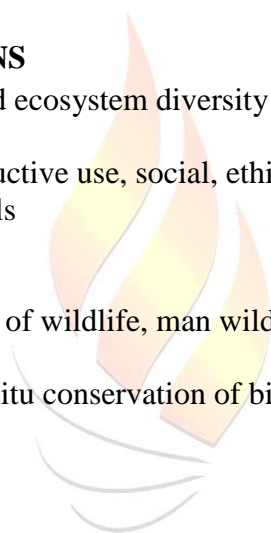
#### **UNIT IV**

##### **BIODIVERSITY AND ITS CONSERVATIONS**

- A. Introduction – Definition: genetic, species and ecosystem diversity
- B. Biogeographical classification of India
- C. Value of biodiversity: consumptive use, productive use, social, ethical aesthetic and option values.
- D. Biodiversity at global, national and local levels
- E. India as a mega-diversity nation
- F. Hot-spots of biodiversity
- G. Threats to biodiversity: habitat loss, poaching of wildlife, man wildlife conflicts
- H. Endangered and endemic species of India
- I. Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

#### **UNIT V: ENVIRONMENTAL POLLUTION**

- A. Definition
- B. Causes, effects and control measures of:
  - Air pollution
  - Water pollution
  - Soil pollution
  - Marine pollution
  - Noise pollution
  - Thermal pollution
  - Nuclear pollution
- C. Solid waste management: Causes, effects and control measures of urban and industrial wastes.
- D. Role of an individual in prevention of pollution
- E. Pollution case studies
- F. Disaster management: floods, earthquake, cyclone and landslides



<b>BCS-152</b>	<b>DATA STRUCTURES USING C LAB</b>	<b>L-T-P</b>	<b>Cr</b>
		<b>0-0-2</b>	<b>1</b>

### **COURSE OUTCOMES**

**CO1** Be able to design and analyze the time and space efficiency of the data structure

**CO2** Be capable to identify the appropriate data structure for given problem

**CO3** Have practical knowledge on the applications of data structures

### **LIST OF EXPERIMENTS/EXERCISES**

1. Write a program to search an element in an array using linear search.
2. Using iteration & recursion concepts write programs for finding the element in the array using Binary Search Method
3. Write a program to perform following operations on matrices using functions only
  - a) Addition
  - b) Subtraction
  - c) Multiplication
  - d) Transpose
4. Write a program for static implementation of Stack
5. Write a program for static implementation of Queue
6. Write a program to create a linked list & perform operations such as insert, delete, update, reverse in the link list
7. Write a program to create a linked list & perform operations such as insert, delete in the Circular link list
8. Write a program to implement binary search tree. (Insertion and Deletion in Binary Search Tree)
9. Write a program which simulates the various tree traversal algorithms.
10. Write a program to implement bubble sort
11. Write a program to implement selection sort
12. Write a program to implement insertion sort
13. Write a program to implement quick sort
14. Write program to implement merge sort

<b>BCS-154</b>	<b>OBJECT ORIENTED PROGRAMMING USING C++ LAB</b>	<b>L-T-P</b>	<b>Cr</b>
		<b>0-0-2</b>	<b>1</b>

### **COURSE OUTCOMES**

**CO1:** Develop solutions for a range of problems using objects and classes.

**CO2:** Programs to demonstrate the implementation of constructors, destructors and operator overloading.

**CO3:** Apply fundamental algorithmic problems including type casting, inheritance, and polymorphism.

**CO4:** Understand generic programming, templates, file handling.

### **LIST OF EXPERIMENTS/EXERCISES**

1. Write a program for Basic/Simple logic building in C++
2. Write a program to implement function overloading
3. Write a program to implement the concept of class and object
4. Write a program to implement the concept of friend function
5. Write a program to implement the concept of static data member
6. Write a program using the concept of constructor & destructor
7. Write a program to Implement operator overloading
8. Write a program to Implement single inheritance
9. Write a program to Implement Multiple inheritance
10. Write a program to Implement Virtual function
11. Write a program to create, read & write sequential file
12. Write a program to create, read & write random access file
13. Write a program to implement function template
14. Write a program to implement class template
15. Write a program for exception handling

### **REFERENCE BOOK**

1. Barkakati, Nabajyoti, "Object Oriented Programming in C++", Prentice Hall of India, 2001.
2. Schildt, Herbert, "C++: The Complete Reference", Tata McGraw Hill, 4<sup>th</sup> Edition, 2003

<b>BCS-158</b>	<b>MATLAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0-0-2</b>	<b>1</b>

## **OBJECTIVE**

MATLAB is a powerful language for technical computing. It is widely used in universities and colleges for courses in mathematics, science and especially in engineering. In industry the software is used in research, development and design. This course is intended for students who are using MATLAB for the first time and have little or no experience in computer programming.

## **COURSE OUTCOMES**

After undergoing this course, the students will be able to:

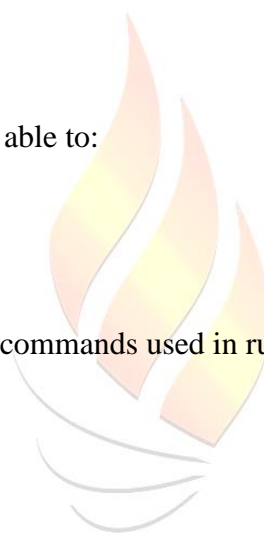
**CO1:** Know the basic concepts of MATLAB

**CO2:** Learn about the arrays as well as strings

**CO3:** Handle the script files and different output commands used in running a script file

**CO4:** State the plotting as well as draw a graph

**CO5:** Handle different function files



## **UNIT I**

**BASIC STRUCTURE AND FEATURES OF MATLAB:** Command window; figure window; editor window and help window; arithmetic operations with scalars, order of precedence; using MATLAB as a calculator; display formats; elementary math built-in functions; scalar variables, assignment operator; predefined variables; useful commands for managing variables; applications in problem solving.

## **UNIT II**

**CREATING ARRAYS:** One dimensional, two dimensional; array addressing; built-in functions for handling arrays; mathematical operations with matrices; strings and strings as variables; generation of random numbers; examples of MATLAB applications.

## **UNIT III**

**SCRIPT FILES:** Creating and saving a script file, current directory; output commands

## **UNIT IV**

**TWO – DIMENSIONAL PLOTS:** Plot command; line specifiers plot of a given data; plot of a function; plotting multiple graphs in the same plot.

## **UNIT V**

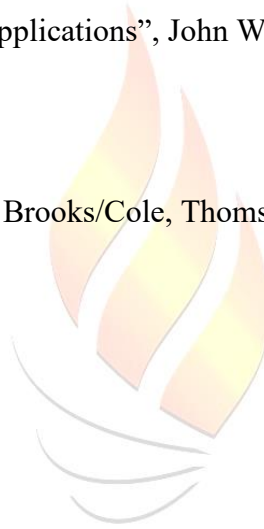
**FUNCTIONS AND FUNCTION FILES:** Creating a function file; input and output arguments; function body; comment lines; saving a function files; using a function file; programming in MATLAB.

### **TEXT BOOK**


Gilat Amos, “MATLAB: An Introduction with Applications”, John Wiley & Sons, Inc (Wiley Student Edition), 2008

### **REFERENCE BOOK**

Herniter, E. Marc, “Programming in MATLAB”, Brooks/Cole, Thomson Learning



**Syllabus**  
**of**  
**BSC**  
**2<sup>nd</sup> Year**  
**3<sup>rd</sup> Semester**



BCS-201	OPERATING SYSTEMS	L T P	Cr
		4-0-0	4

## OBJECTIVE

To provide the knowledge of internals, different types and purpose of operating systems

## PRE-REQUISITES

Knowledge of computer organization and architecture, programming skills

## COURSE OUTCOMES

The students undergoing this course will be able:

**CO1:** To get familiar with the basic concepts of operating system

**CO2:** To know about the multiprocessing, synchronization & deadlocks

**CO3:** To learn the strategy to manage the memory available

**CO4:** To learn the concepts of files, their accession and disk scheduling

**CO5:** To know about the hardware devices and drivers used for them

## UNIT I

**PROCESS MANAGEMENT:** Functionalities and Objectives of OS, historical evolution of operating systems, types of operating system, O/S service system calls, system programs, Process States, Process Control Block, Processor Scheduling, CPU scheduling, scheduling criteria, scheduling algorithms

## UNIT II

**PROCESS-SYNCHRONIZATION & DEADLOCKS:** Critical Section; Mutual exclusion, Process cooperation,; Deadlocks: deadlock prevention; avoidance and detection; deadlock recovery; Dining philosophers problem; semaphores.

## UNIT III

**MEMORY MANAGEMENT:** Logical & Physical Address Space; swapping; contiguous memory allocation; non-contiguous memory allocation paging and segmentation techniques; segmentation with paging; virtual memory management - Demand Paging & Page-Replacement Algorithms;

## UNIT IV

**FILE SYSTEM:** Different types of files and their access methods; directory structures; various allocation methods; disk scheduling and management.

## UNIT V

**I/O SYSTEMS:** I/O Hardware; Device Controllers; Interrupt Handlers; Device Drivers; Application I/O Interface; Kernel; Transforming I/O requests; Performance Issues.

## **TEXT BOOK**

Silberchatz et al, —Operating System Concepts, 5th edition, Addison-Wesley, 1998

## **REFERENCE BOOKS**

1. Tanenbaum A., —Modern Operating Systems, Prentice-Hall, 1992
2. Stallings William, —Operating Systems Internals and Design Principles, 4th edition, Prentice-Hall, 2001
3. Dhamdhare D. M., —Operating System, 2nd Edition, Tata McGraw Hill, 1999
4. Kernighan Brian and Pike Rob, —The Unix Programming Environment, Prentice Hall of India, 1984
5. Bach Maurich, —Design of the Unix Operating System, Prentice Hall of India, 1986
6. Muster John, —Introduction to UNIX and LINUX, Tata McGraw Hill, 2003
7. Ritchie Colin, —Operating System Incorporating Unix & Windows, Tata McGraw Hill, 1974
8. Madnick Stuart and Donovan John, —Operating Systems, Tata McGraw Hill, 2001

BCS-203	COMPUTER ARCHITECTURE AND ORGANIZATION	L T P	Cr
		4-0-0	4

## **OBJECTIVE**

To provide basic knowledge of internals of microprocessor, its architecture, components, terminologies, etc. at minute level and ultimately about the working of a digital computer hardware as a whole.

## **PRE-REQUISITES**

Knowledge of data structures, microprocessors and interfacing

## **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To learn the basic architecture of the computer

**CO2:** To learn about the different addresses and instructions passed for

**CO3:** To know about the different types of architecture & instruction cycle

**CO4:** To get knowledge about the different types of memories and their hierarchies

**CO5:** To know about the parallel processing

## **UNIT I**

**GENERAL SYSTEM ARCHITECTURE:** k-map, number system, FF, Boolean algebra, design of logic gates. Functions and block diagram of computer, store program control concept, Flynn's classification of computers (SISD, MISD, MIMD); multilevel viewpoint of a machine: digital logic, operating systems, high level language; structured I/O; performance metrics; MIPS, MFLOPS, GFLOPS and TFLOPS.

## **UNIT II**

**INSTRUCTION SET ARCHITECTURE:** Instruction codes, instruction set formats (fixed, variable, hybrid); types of instructions, memory reference, register reference, I/O reference; addressing modes: register, immediate, direct, indirect, indexed; operations in the instruction set; arithmetic and logical, data transfer, control flow; types of interrupts; timing and control; instruction set based classification of processors (RISC, CISC, and their comparison).

## **UNIT III**

**BASIC NON PIPELINED CPU ARCHITECTURE:** CPU Architecture types (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU, fetch-decode-execute cycle (typically 3 to 5 stage); micro-instruction formats, implementation of control unit: hardwired and micro-programmed, control memory, microinstruction sequencing.

## **UNIT IV**

**MEMORY HIERARCHY & I/O TECHNIQUES:** Need for a memory hierarchy (Locality of Reference Principle, memory hierarchy in practice: cache, main memory and secondary memory, memory parameters: access/ cycle time, cost per bit); main memory (semiconductor RAM & ROM organization, memory expansion, static & dynamic memory types); cache memory: associative & direct mapped cache organizations. Internal memory, High speed memory.

## **UNIT V**

**INTRODUCTION TO PARALLELISM:** Goals of parallelism (exploitation of concurrency, throughput enhancement); Amdahl's law; instruction level parallelism (pipelining, super scaling-basic features); processor level parallelism (multiprocessor systems overview), parallel processing concept, processing power and buses of a microprocessor, components of microprocessor; I/O ports.

## **TEXT BOOK**

1. Carpinelli, —Computer Organization & Architecture| Tata McGraw Hill, 2001

## **REFERENCE BOOKS**

1. Stallings. W, —Computer Organization & Architecture: Designing For Performance, 6th Edition, Prentice Hall of India, 2002/ Pearson Education Asia, 2003



2. Mano M Morris, —Computer System Architecture, 3rd Edition, Prentice Hall of India Publication, 2001 / Pearson Education Asia, 2003
3. Rajaraman V. and Radhakrishnan T, —Introduction to Digital Computer Design, 4th Edition, Prentice Hall of India 2004.
4. Stalling William, —Computer Organization and Architecture, 7th Edition, Prentice Hall of India, 2005.



BCS-205	MULTIMEDIA TECHNOLOGIES	L T P	Cr
		4-0-0	4

### OBJECTIVE

To provide basic knowledge of image compression, audio, video, sound, virtual reality, intelligent multimedia systems etc.

### PRE-REQUISITES

Knowledge of computer graphics, programming, 3D geometry

### COURSE OUTCOMES

The students undergoing this course will be able:

**CO1:** To get familiar with different types of media

**CO2:** To know about the network of multimedia distribution as well as animations

**CO3:** To learn about different types of images, color models etc.

**CO4:** To learn about different signals and digital sound

**CO5:** To know how to add motion in pictures leads to a video

### UNIT I

**BASICS OF MULTIMEDIA TECHNOLOGY:** Introduction and application of multimedia, framework for multimedia systems; multimedia authoring tools, concept of multimedia project, stages, interactive and non-interactive multimedia, different storage device

### UNIT II

**MULTIMEDIA Distribution:** Internet; World Wide Web & multimedia distribution network: ATM & ADSL; multimedia servers & databases;

**Animation:** Introduction types of animation, morphing, color modes, importance of text, Unicode.

### UNIT III

**IMAGE COMPRESSION & STANDARDS:** Making still images; editing and capturing images; scanning images; computer color models; color palettes; vector drawing; 3D drawing and rendering; JPEG: objectives and architecture; DCT encoding and quantization; statistical coding; predictive lossless coding; performance; overview of other image file formats as GIF; TIFF; BMP; PNG; etc.

### UNIT IV

**AUDIO:** Digital representation of sound; time domain sampled representation; method of encoding the analog signals; sub-band coding; Fourier method; transmission of digital sound; digital audio signal processing; stereophonic and quadraphonic signal processing; editing sampled sound; MPEG Audio; audio compression and decompression; brief survey of speech recognition and generation; audio synthesis; musical instrument digital interface

### UNIT V

**VIDEO:** digital video and image compression; MPEG motion video compression standard; DVI technology; time base media representation and delivery. Video on demand. **VIRTUAL REALITY:** Applications of multimedia; intelligent multimedia system; desktop virtual reality; VR operating system; virtual environment displays and orientation making; visually coupled system requirements; intelligent VR software systems.

### **TEXT BOOK**

Villamil and Molina, "An Introduction to Multimedia", MacMillan, 1997

### **REFERENCE BOOKS**

1. Lozano, “Multimedia: Sound & Video”, Prentice Hall of India (Que), 1997
3. Ranjan Parekh, “Principle of Multimedia”, Tata McGraw Hill
4. Villamil and Molina, “Multimedia: Production, Planning and Delivery”, Que, 1997
5. Sinclair, “Multimedia on the PC”, BPB Publications
6. Tay Vaughan, “Multimedia: Making It Work”, Fifth edition, Tata McGraw Hill, 1994
7. James E Shuman, “Multimedia in Action”, Wadsworth Publications, 1997
8. Jeff Coate Judith, “Multimedia in Practice”, Prentice Hall of India, 1995
9. John F. Koegel, “Multimedia Systems”, Addison Wesley Ltd.



BCS-207	COMPUTER NETWORKS	L T P	Cr
		3-0-0	3

### **OBJECTIVE**

To have a fundamental understanding of the design, performance and state of the art of wireless communication systems, Topics covered include state of the art wireless standards and research and thus changes substantially form one offering of this course to the next

### **PRE-REQUISITES**

Knowledge of computers hardware and software

### **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To know different types of networks and basic architecture of network

**CO2:** To learn about different network models

**CO3:** To know deeply about the local area network

**CO4:** To know deeply about the wide area network

**CO5:** To aware about application layer of network models

### **UNIT I**

**OSI REFERENCE MODEL AND NETWORK ARCHITECTURE:** Introduction to Computer Networks; Example networks ARPANET; Internet; Private Networks; Network Topologies: Bus-; Star-; Ring-; Hybrid-; Tree -; Complete -; Irregular –Topology, Types of Networks: Local Area Networks; Metropolitan Area Networks; Wide Area Networks; Layering architecture of networks; OSI model; Functions of each layer; Services and Protocols of each layer.

### **UNIT II**

**TCP/IP:** Introduction; History of TCP/IP; Layers of TCP/IP; Protocols; Internet Protocol; Transmission Control Protocol; User Datagram Protocol; IP Addressing; IP address classes; Subnet Addressing; Internet Control Protocols; ARP; RARP; ICMP; Application Layer; Domain Name System; Email – SMTP; POP; IMAP; FTP; NNTP; HTTP; Overview of IP version 6.

### **UNIT III**

**LOCAL AREA NETWORKS:** Introduction to LANs; Features of LANs; Components of LANs; Usage of LANs; LAN Standards; IEEE 802 standards; Channel Access Methods; Aloha; CSMA; CSMA/CD; Token Passing; Ethernet; Layer 2 & 3 switching; Fast Ethernet and Gigabit Ethernet; Token Ring; LAN interconnecting devices: Hubs; Switches; Bridges; Routers; Gateways.

### **UNIT IV**

**WIDE AREA NETWORKS:** Introduction of WANs; Routing; Congestion Control; Quality of Service, WAN Technologies; Synchronous Digital Hierarchy (SDH)/ Synchronous Optical Network (SONET); Asynchronous Transfer Mode (ATM); Frame Relay.

### **UNIT V**

**APPLICATION LAYER:** Remote Logging, Electronic Mail, SMTP, POP, IMAP FTP, WWW, HTTP, SNMP, Network Security: Services, Message Confidentiality, Authentication, Integrity, Firewalls.

### **TEXT BOOK**

1. Forouzan Behrouz A., “Data Communications and Networking”, Tata McGraw Hill 2006.

### **REFERENCE BOOKS**

1. Tanennbaum Andrew S, “Computer Networks”, 4th Edition, Pearson Education/Prentice Hall of India, 2003.
2. Stallings William, “Data and Computer Communication”, 5th Edition, Prentice Hall of India, 1999



BCS-209	CORE JAVA	L T P	Cr
		4-0-0	4

### OBJECTIVE

To relay the theoretical and practical knowledge of Java programming language

### PRE-REQUISITES

Basic Knowledge of programming language and object oriented programming

### COURSE OUTCOMES

The students undergoing this course will be able:

**CO1:** To learn about the basics of objects, object behavior & storage of different objects

**CO2:** To know about the basics of Java programming language

**CO3:** To learn how to create a package as well as handling of exceptions

**CO4:** To learn about the multithreading as well as synchronization of threads to avoid deadlocks

**CO5:** To aware about the database connectivity using JDBC as well as other APIs

### UNIT I

**INTRODUCTION TO JAVA & PRINCIPLES OF OBJECT ORIENTED PROGRAMMING:** Basic Concepts of OOP and its Benefits, Application of OOP, Features of Java. Data types & Operators available in java; Control Structures: if, while, do while, for, switch; Break & Continue Statement; Arrays and Strings: Arrays, Arrays of Characters; String handling Using String Class; Operations on String Handling Using; String Buffer Class.

### UNIT II

**OBJECT-ORIENTATION:** Object-Oriented Programming in Java, Java Program Structure. Defining of a Class, Definition of Methods, Constructors, Creating Objects of a Class, Assigning Object Reference Variables, The keyword "this", Defining and Using a Class, Automatic Garbage Collection. Extending Class and Inheritance: Using Existing Classes, Class Inheritance, Choosing Base Class, Access Attributes, Polymorphism, Multiple Levels of Inheritance, Abstraction through Abstract Classes, Using Final Modifier, the Universal Super class-Object Class.

### UNIT III

**PACKAGE & EXCEPTION:** Understanding Packages, Defining a Package, Packaging up your Classes, Adding Classes from a Package to your Program, Understanding CLASSPATH, Standard Packages, Access Protection in Packages, Concept of Interface. Exception Handling: The Idea behind Exceptions, Types of Exceptions, Dealing with Exceptions, Exception Objects, Defining Your Own Exceptions, Checked and Unchecked Exceptions.

### UNIT IV

**MULTITHREADING PROGRAMMING:** The Java Thread Model, Understanding Threads, The Main Thread, Creating a Thread: extending Thread and implementing Runnable, Creating Multiple Threads, Thread Priorities, Synchronization, Deadlocks inter-thread communication, Deadlocks. Input/Output in Java: I/O Basic, Byte and Character Structure, I/O Classes, Reading Console Input, Writing to Console Output, Reading and Writing on Files, Random Access Files, Storing and Retrieving Objects from File. Stream Benefits.

### UNIT V

**JAVA DATA BASE CONNECTIVITY (JDBC):** Database Connectivity- Relation Databases; JDBC API; Reusing Database Objects; Transactions; Advance Techniques. Java Utilities (java.util Package) The Collection Framework : Collections of Objects; Collection Types; Sets Sequence Map; Understanding Hashing; Use of Array List & Vector.

### **TEXT BOOK:**

Balaguruswamy , E., ““Programming with Java”, Tata Mcgraw Hill.

### **REFERENCE BOOKS**

1. Horetmann Cay and Cornell Gary, “Core Java™ 2, Volume II – Advanced Features”, 7th Edition, Pearson Publisher, 2004.
2. Horetmann Cay and Cornell Gary, “Core Java Volume – I”, Pearson Education.
3. Callway Dustin R., “Inside Servlets”, Pearson Education.
4. Goodwill James and Bryan Morgan, “Developing Java Servlets”, Techmedia.
5. “Java Server Programming, Volume I and II”, Wrox Press



BCS-211	DATABASE MANAGEMENT SYSTEM	L-T-P	Cr
		4-0-0	4

### **OBJECTIVE**

To provide knowledge about various organizations and management information systems, keeping in view the aspects of shareability, availability, evolvability and integrity.

### **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To know the basics of database & its architecture

**CO2:** To aware about the existing data models, entities as well as constraints

**CO3:** To learn about the different anomalies of the data and ways to normalize it

**CO4:** To know about the storage of data in the files & organization of files

**CO5:** To learn about the transactions and its ways done on the database

### **UNIT I**

**INTRODUCTION:** Purpose of database system; characteristics of database approach; advantages of using DBMS; database concept and architecture; data abstraction, data models; instances and schema; data independence; schema architecture; database languages; database manager; database administrator; database users.

### **UNIT II**

**DATA MODELING:** Introduction to Hierarchical model, Network model. Relational model, E-R Model, Entity sets attributes and keys; relationships (ER); database modelling using entity; type role and structural constraints; weak and strong entity types; entity-relationship diagram-basic concepts; Enforcing Data Integrity Constraints; Relational-Algebra Operations; Introduction on views; Codd's Rules.

### **UNIT III**

**NORMALIZATION& SQL:** Database design process; relational database design; relation schema; anomalies in a database; functional dependencies; 1NF, 2NF, 3NF and BCNF. ; Reduction of an E-R schema to Tables; Introduction to SQL; basic queries in SQL; advanced queries in SQL; functions in SQL; basic data retrieval; updates in SQLs, views in SQL.

### **UNIT IV**

**FILE ORGANIZATION:** indexing and hashing; overview of file organization techniques; secondary storage devices; operations in files; heap files and sorted files; ; Indexing and Hashing- Basic concepts; Static Hashing; Dynamic Hashing; ordered indices; single level ordered index; multi-level index.

### **UNIT V**

**TRANSACTION PROCESSING & QUERY PROCESSING:** Desirable properties of transactions; implementation of atomicity and durability; schedules and recoverability; serializability of schedules; concurrency control. Deadlock handling - detection and resolution.

### **TEXT BOOK**

1. Silberschatz A., Korth H. F. and Sudarshan S., “Database System Concepts”, 3rd edition, McGraw-Hill, International Edition, 1997

### **REFERENCE BOOKS**

1. Date C. J., “An Introduction to Database Systems”, 7th edition, Addison- Wesley, Low Priced Edition, 2000
2. Desai Bipin, “Introduction to Database Management System”, Galgotia Publications, 1991



3. Elmasri R. and Navathe S. B., “Fundamentals of Database Systems”, 3rd edition, Addison-Wesley, Low Priced Edition, 2000



BCS-255	MULTIMEDIA TECHNOLOGIES LAB	L T P	Cr
		0-0-2	1

### **OBJECTIVE**

To provide practical knowledge of concepts of different medias.

### **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To get familiar with different types of media

**CO2:** To implement the motion in still images

**CO3:** To learn about shape tweening of objects & alphabets

**CO4:** To use the macromedia flash

**CO5:** To know how to add text in the file

### **List of Experiments**

1. Create Motion Tweening using Macromedia Flash (Moving Ball).
2. Design a moving ball in V-shape
3. Design a moving ball in W-shape
4. Design shape tweening – Object to Object
5. Design shape tweening – Object to Alphabet
6. Design shape tweening – Alphabet to Object
7. Create Spot Light using Macromedia Flash.
8. Implement Splitting of Ball
9. Create Animated Pool Table.
10. Create Bow & Arrow hitting a Ball.
11. Write any Text with the illusion of pen writing letters.



BCS-259	CORE JAVA LAB	L T P	Cr
		0-0-2	1

### OBJECTIVE

To relay the practical knowledge of Java programming language

### COURSE OUTCOMES

The students undergoing this course will be able:

CO1: To learn about operators used in Java Programming

CO2: To know about loops implemented in Java programming language

CO3: To implement the arrays and strings

CO4: To learn about the inheritance, packages & exception handling

CO5: To aware about the database connectivity using JDBC

### LIST OF EXPERIMENTS

The following programs on different topic are to be done in this lab.

#### 1. Sample Program

(a) Write a Java program to print “Hello Java”

#### 2. Operators and Expressions

(a) Write a java program to find the area of a rectangle.

(b) To write a java program to find the result of the following expressions

(i)  $(a \ll 2) + (b \gg 2)$

(ii)  $(b > 0)$

(iii)  $(a + b * 100) / 10$

(iv)  $a \& b$

Assume  $a=10, b=5$

(c) To write a java program to print the individual digits of a 3 digit number using Command line arguments.

#### 3. Decision making statements

(a) Write a java program to read two integers and print the larger number. followed by the words “is larger”. If the numbers are equal print the message “These numbers are equal”

(b) Write a java program to read an integer and find whether the number is odd or even.

(c) Write a java program to find the number of and sum of all integers greater than 100 and less than 200 that are divisible by 7.

#### 4. Looping Statements

(a) Write a Java program to find the sum of digits of a given number.

(b) Write a java program to find the first 15 terms of Fibonacci sequence.

(c) Write a java program to print the Armstrong numbers.

(d) Given a number, write a program using while loop to reverse the digits of the number.

For example, the number

12345

should be written as 54321.

#### 5. Array & Strings

(a) Write a java program to find the largest and smallest number in an array.

(b) Write a java program to multiply two matrices.

(c) Write a java program to sort the following numbers in descending order.

{55, 40, 80, 65, 71}

(d) Write a java program that creates a string object and initializes it with your name and performs the following operations

(i) To find the length of the string object using appropriate String method.

- (ii) To find whether the character 'a' is present in the string. If yes find the number of times 'a' appear in the name and the location where it appears.
- (e) Write a java program to arrange the following word in alphabetical order  
{Madras, Delhi, Ahmadabad, Calcutta, Bombay}
- (f) Write a java program to create a StringBuffer object and illustrate how to append characters and to display the capacity and length of the StringBuffer.
- 6. Classes & Objects**
- (a) Write a java program to display total marks of 5 students using student class. Given the following attributes: Regno(int), Name(string), Marks in subjects(Integer Array), Total (int).
- (b) Write a java program to find the area of a room using constructor.
- (c) Write a java program to implement method overloading.
- (d) Write a java program to show the use of "static" members.
- (e) Write a java program to implement the nesting of methods.
- 7. Inheritance**
- (a) Write a java program to implement single inheritance using "super" keyword.
- (b) Write a java program to implement method overriding.
- (c) Write a java program to implement multiple inheritances.
- 8. Package & Multithreading**
- (a) Write a program to create your own package and use that package in another program to print "Hello package".
- (b) Write a program to implement multithreading using the system function like yield(), stop(), sleep().
- 9. Exception Handling & File handling**
- (a) Write a java program to implement multiple try/catch statements.
- (b) Write a program to copy the content of one file into another using character stream classes.
- (c) Write a program to copy the content of one file into another using byte stream classes
- 10. Database Connectivity**
- (a) Write a programme to connect java application GUI with database.(JDBC)
- (b) Write a program to select all the information of a table named as "Student".

### **TEXT BOOK**

Herbert Schildt , "The Complete Reference Java 2 fifth edition, McGraw Hill.

### **REFERENCE BOOKS**

1. Balaguruswamy , E., ""Programming with Java", Tata Mcgraw Hill.
2. Horetmann Cay and Cornell Gary, "Core Java Volume – I", Pearson Education.

BCS-261	DATABASE MANAGEMENT SYSTEM LAB	L-T-P	Cr
		0-0-2	1

### OBJECTIVE

To provide knowledge about implementation of practical aspects of database i.e. creation of tables and applying queries using SQL queries

### COURSE OUTCOMES

The students undergoing this course will be able:

**CO1:** To know the basics of structured query language

**CO2:** To aware about applying different queries on database structured in the form of tables

**CO3:** To learn about the different SQL queries performed using operators as well as constraints

**CO4:** To create views from created table to further organize the data


**CO5:** To learn about the basic operations of relational algebra

### LIST OF EXPERIMENTS/EXERCISES

1. Introduction to SQL.
2. Write a query for:
  - (i) Creation of table.
  - (ii) Insertion of data into table
  - (iii) Displaying the data of table.
  - (iv) Deletion of data from table
  - (v) Updating the data
  - (vi) Modifying the structure of a table.
3. Finding unique names of all salesmen, deletion of the structure of a table, use of delete command with conditions, updating records of a table with conditions, altering structure of a table and changing size of existing column in the table
4. Arithmetic operators, logical operators and pattern matching operator.
5. Key constraints: primary key constraints, foreign key constraints, not null constraints and unique constraints; use of check constraints.
6. Aggregate and mathematical functions: count, count(\*), Avg, max, min, sum, lower, upper, power , sqrt.
7. Creating views from single and multiple tables, drop views and creating index on the table and drop them.
8. Binary operations in Relational Algebra: Union, Intersection, Set Difference, Join, Cartesian product.
9. Grouping of data into tables and listing records in ascending order or descending order.
10. Creation of sequences and explain use of sequences.
11. Access permissions in SQL.



Syllabus  
Of  
BSC  
2<sup>nd</sup> Year  
4<sup>th</sup> Semester



<b>BCS-202</b>	<b>COMPUTER GRAPHICS</b>	<b>L T P</b>	<b>Cr</b>
		<b>4 0 0</b>	<b>4</b>

## **OBJECTIVES**

- Write programs Using C/C++/ OpenGL graphics environment.
- Use polygonal and other modelling methods to describe scenes.
- Understand and be able to apply geometric transformations.
- Create basic animations.

## **PRE-REQUISITES**

Knowledge of computer programming

## **COURSE OUTCOMES**

Students after undergoing this course will be able to:

**CO1:** Understand the basics of computer graphics, different graphics systems and applications of computer graphics.

**CO2:** Discuss various algorithms for scan conversion and filling of basic objects and their comparative analysis.

**CO3:** Use of geometric transformations on graphics objects and their application in composite form.

**CO4:** Extract scene with different clipping methods and its transformation to graphics display device.

**CO5:** Explore projections and visible surface detection techniques for display of 3D scene on 2D screen.

## **UNIT I**

**INTRODUCTION:** What is computer graphics, computer graphics applications, computer graphics hardware and software, basic graphics system and standards; Raster Scan and Random Scan graphics; Raster-Scan Display System, Video Controller, Random-Scan Display processor, frame buffer.

## **UNIT II**

**ALGORITHMS:** Two dimensional graphics primitives: points and lines, line drawing algorithms: DDA, Bresenham's; circle drawing algorithms: using polar coordinates, Bresenham's circle drawing, midpoint circle drawing algorithm.

## **UNIT III**

**TWO AND THREE DIMENSIONAL TRANSFORMATION:** Two dimensional transformations: translation, scaling, rotation, reflection, shearing, transformation, Homogeneous coordinates system, 2D composite transformation: matrix Representation of 3-D transformations, composition of 3-D transformation;

## **UNIT IV**

**TWO DIMENSIONAL VIEWING AND CLIPPING:** The 2-D viewing pipeline, windows, viewports, window to View port mapping; clipping: point, clipping line (algorithms): 4 bit code algorithm, Sutherland-Cohen algorithm, parametric line clipping algorithm (Cyrus Beck). Sutherland-Hodgeman polygon clipping algorithm

## **UNIT V**

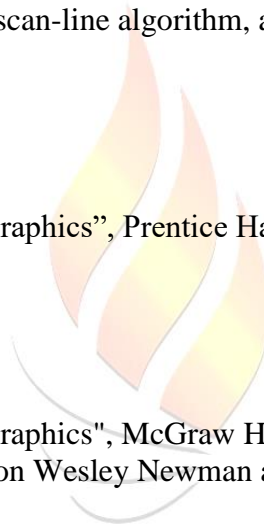
**THREE DIMENSION VIEWING AND HIDDEN SURFACE REMOVAL:** viewing in 3D: projections, types of projections; the mathematics of planner geometric projections; coordinate systems: Introduction to hidden surface removal; the Z- buffer algorithm, scan-line algorithm, area sub-division algorithm.

## **TEXT BOOK**

Donald Hearn & M. Pauline Baker, "Computer Graphics", Prentice Hall of India

## **REFERENCE BOOK**

1. Rogers, "Procedural Elements of Computer Graphics", McGraw Hill
2. Asthana, Sinha, "Computer Graphics", Addison Wesley Newman and Sproul, "Principle of Interactive Computer Graphics", McGraw Hill
3. Steven Harrington, "Computer Graphics", A Programming Approach, 2nd Edition
4. Rogar and Adams, "Mathematical Elements of Computer Graphics", McGraw Hill.





<b>BCS-204</b>	<b>DESIGN &amp; ANALYSIS OF ALGORITHMS</b>	<b>L T P</b>	<b>Cr</b>
		<b>4 0 0</b>	<b>4</b>

**OBJECTIVE**

To relay the theoretical and practical aspects of design of algorithms.

**PRE-REQUISITES**

Knowledge of fundamentals of basic computer programming for implementing algorithms.

**COURSE OUTCOMES**

**CO1:** Define the basic concepts of algorithms and analyze the performance of algorithms.

**CO2:** Discuss various algorithm design techniques for developing algorithms.

**CO3:** Discuss various searching, sorting and graph traversal algorithms.

**CO4:** Understand NP completeness and identify different NP complete problems.

**CO5:** Discuss various advanced topics on algorithms.

**UNIT I**

**MATHEMATICAL PRELIMINARIES:** Role of algorithm in computing; mathematical preliminaries; review of growth functions; solution of difference equations; analysis in terms of space and time complexity.

**UNIT II**

**SORTING:** Insertion sort; merge sort; heap sort; quick sort; radix sort, bucket sort, bubble sort, selection sort.

**UNIT III**

**ADVANCED DATA STRUCTURES:** Stack; queue; linked list; binary search trees; red black trees.

**UNIT IV**

**DYNAMIC PROGRAMMING AND GREEDY ALGORITHM:** Matrix multiplications; longest common subsequence, Huffman coding; task scheduling problem.

**UNIT V**

**ELEMENTARY GRAPH ALGORITHM AND MINIMUM SPANNING TREE:** Representation of graph; breadth-first search; depth first search; topological sort; strongly connected components, Growing a minimum spanning tree; Kruskal & Prims algorithms.

**TEXT BOOK**

Cormen Thomas H., Leiserson Charles E. and Rivest Ronald L., “Introduction to Algorithms”, Tata McGraw Hill, 1990.

## **REFERENCE BOOKS**

1. V. Aho, J. E. Hopcroft, J. D. Ullman, “The Design and Analysis of Computer Algorithms”, Addison Wesley, 1998.
2. Ellis Horowitz and Sartaz Sahani, “Computer Algorithms”, Galgotia Publications, 1999.
3. E. Knuth, “The Art of Computer Programming”, 2<sup>nd</sup> Ed., Addison Wesley, 1998



BCS-206	LINUX AND SHELL PROGRAMMING	L-T-P	Cr
		3-0-0	3

## OBJECTIVE

The objective of the course aims to introduce about open source operating system as We can use Linux as Server OS or as standalone OS on our PC, Shell scripting & IPC etc.

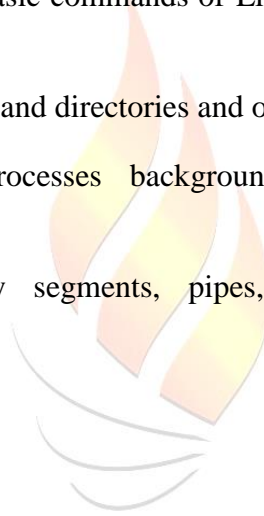
## COURSE OUTCOMES

**CO1:** Students will be able to understand the basic commands of Linux operating system and can write shell scripts

**CO2:** Students will be able to create file systems and directories and operate them

**CO3:** Students will be able to create processes background and fore ground etc.. by fork() system calls

**CO4:** Students will create shared memory segments, pipes, message queues and can exercise inter-process communication



## UNIT I

**UNIX UTILITIES:** Introduction to UNIX file system; vi editor; file handling utilities; security by file permissions; process utilities; disk utilities; networking commands; cp; mv; ln; rm; unlink; mkdir; rmdir; du; df; mount; umount; find; ps; who; w; finger; arp; ftp; telnet; rlogin; text processing utilities and backup utilities; detailed commands to be covered are cat; tail; head; sort; nl; uniq; grep; egrep; fgrep; cut; paste; join; tee; pg; comm.; cmp; diff; tr; awk; tar; cpio.

## UNIT II

**PROBLEM SOVING APPROACHES IN UNIX:** Using single commands; using compound commands; shell scripts; C programs; building own command library of programs; working with the Bourne shell: what is a shell; shell responsibilities; pipes and input redirection; output redirection; shell script examples.

## UNIT III

**UNIX FILES:** UNIX file structure; directories; files and devices; system calls; library functions; usage of open; creat; read write; close; lseek; stat; fstat; ioctl; umask; dup; dup2; the standard I/O ( fopen; fclose; fflush; fseek; fgetc; getc; getchar; fputc; putc; putchar; fgets; gets); formatted I/O; stream errors; streams and file descriptors; file and directory maintenance (chmod; chown; unlink; link; symlink; mkdir; rmdir; chdir; getcwd).

## UNIT IV

**UNIX PROCESS AND SIGNALS AND INTER-PROCESS COMMUNICATION:** what is process; process structure; starting new process; waiting for a process; zombie process; process control; process identifiers; system call interface for process management-fork; vfork; exit; wait; waitpid; exec; system.

**INTRODUCTION TO IPC;** IPC between processes on a single computer system; IPC between process on different systems; file and record locking; other UNIX locking techniques; pipes; FIFO; streams and messages

## **UNIT V**

**MESSAGE QUEUES AND SEMAPHORES:** UNIX system-V messages; UNIX kernel support for message; UNIX APIs for messages client/server example.UNIX system-V semaphores; UNIX kernel support for semaphores; Unix APIs for semaphores; Shared Memory- UNIX system-V shared memory.

## **TEXT BOOKS**

W. R. Stevens, “Unix Network Programming”, Pearson/PHI.

## **REFERENCE BOOKS**

Sumitabha Dass,”Unix Concepts and Application”, 3<sup>rd</sup> Edition, Tata McGraw Hill.



<b>BS-220</b>	<b>APPLIED NUMERICAL TECHNIQUES</b>	<b>L-T-P</b>	<b>Credits</b>
		<b>3-0-0</b>	<b>3</b>

### **OBJECTIVE**

To acquaint the students with the various concepts and tools of applied mathematics which will be very basic and the very soul and guide of various engineering subject.

### **COURSE OUTCOMES**

Students undergoing this course will be able to:

**CO1:** It is used for solving a system of equations

**CO2:** To know how to find the roots of transcendental equations.

**CO3:** To learn how to interpolate the given set of values

**CO4:** To understand the curve fitting for various polynomials

**CO5:** To learn numerical solution of differential equations

### **UNIT I**

**SOLUTION OF NONLINEAR EQUATIONS :** Introduction to numbers and their accuracy; absolute, relative and percentage errors and their analysis; Bisection method ; Regula- falsi method; secant method; fixed point iteration method; Newton- Raphson method; convergence criteria of methods.

### **UNIT II**

**SOLUTION OF SIMULTANEOUS LINEAR EQUATIONS :** Gauss elimination method; Gauss-Jordan method; UV factorization method; Jacobi's iteration method; Gauss-Seidal iteration method; .

### **UNIT III**

**INTERPOLATION AND CURVE FITTING:** Introduction to interpolation; Newton's forward and backward interpolation formulae; Gauss's forward and backward interpolation formulae; Stirling formula; Lagrange interpolation; Newton's divided difference formula; Principle of least squares; curve fitting by least square method

### **UNIT IV**

**NUMERICAL DIFFERENTIATION AND INTEGRATION:** Numerical differentiation formulae: differentiation by using forward interpolation formula; backward interpolation formula; Stirling formula; Newton-Cotes formula for numerical integration: Trapezoidal rule; Simpson's rules; Romberg' method.

## **UNIT V**

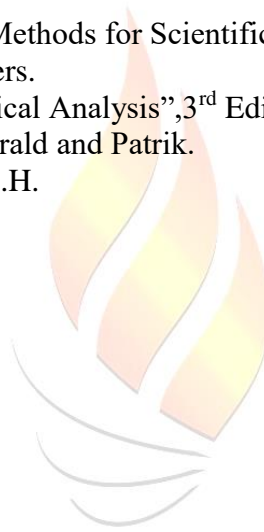
**SOLUTION OF ORDINARY DIFFERENTIAL EQUATION** : ,Taylor series method; Euler method; Euler modified method; Runge kutta method; Milne's predictor -corrector method; Adams-Bashforth method for finding solution of differential equation.

### **TEXT BOOK**

Grewal, B. S., "Numerical methods in Engineering and Science",9<sup>th</sup> Edition, 2010,Khanna publishers.

### **REFERENCE BOOKS**

1. Jain, R.K. and Iyengar, S.R.K., "Numerical Methods for Scientific and Engg. Computations" ,5<sup>th</sup> Edition,2007, New Age International publishers.
2. Sastry, S.S.,"Introductory Methods of Numerical Analysis",3<sup>rd</sup> Edition,1999, Prentice Hall of India.
3. Applied Numerical Analysis" by Curtis F, Gerald and Patrik.
4. Numerical Methods by E. Balagurusamy T.M.H.



<b>BCS-210</b>	<b>IT MANAGEMENT</b>	<b>L T P</b>	<b>CR</b>
		<b>3 0 0</b>	<b>3</b>

## **OBJECTIVE**

The objective of the course aims to introduce about Information technology plays an important role in today's business world. Majority of the companies rely on this for the purpose of data processing, fast communications and acquiring market intelligence. Information technology helps business improve the processes of business it drives revenue growth, helps them achieve cost efficiency and more importantly, ensures they increase revenue growth while maintaining a competitive edge in the market place

## **COURSE OUTCOMES**

**CO1:** Understand the concepts related to Business.

**CO2:** Demonstrate the roles, skills and functions of management.

**CO3:** Analyze effective application of PPM knowledge to diagnose and solve organizational problems and develop optimal managerial decisions.

**CO4:** Understand the complexities associated with management of human resources in the organizations and integrate the learning in handling these complexities.

## **UNIT I**

**INTRODUCTION:** Evolution of computer, Computer basics, Network and internet, computing resources, information technology.

## **UNIT II**

**IT INFRASTRUCTURE:** IT infrastructure management, infrastructure, IT Infrastructure management, challenges in IT infrastructure management, design issues of IT organizations and IT infrastructure, determining customers' requirements, IT systems management process, IT service management process, information system design process, patterns for IT systems management.

## **UNIT III**

**Service Delivery Process & Support Process:** Service level management, financial management, IT service continuity management, capacity management, availability management. **Support process:** Configuration management, incident management, problem management.

## **UNIT IV**

Storage Management: Introduction to storage, backup and storage, archive and re-trieve, disaster recovery, space management, database and application protection, Bare Machine Recovery (BMR), data retention.

## **UNIT V**

Security Management & IT Ethics: Introduction, computer security, internet security, physical security, identity management, access control, intrusion detection ,IT Ethics: Introduction, intellectual property, privacy and law, computer forensics, ethics and internet, cyber crimes

## **TEXT BOOK**

1. Phalguni Gupta, Surya Prakash, Umarani Jayaraman, IT Infrastructure and its Management, Tata McGraw Hill Education Private Limited, ISBN-13: 978-0070699793, 2009.
2. Simon Adams, ITIL V3 Foundation Handbook Updated Edition, Stationery Office Books (TSO) Publisher, 2009.

## **REFERENCES**

1. Ivanka Menken, ITIL V3 Foundation Certification Exam Preparation Course in a Book for Passing the ITIL V3 Foundation Exam, Second Edition (The Art of Service), 2009.
2. Van Haren, Passing the ITIL Foundation, Van Haren Publishing, 2011.



BCS-212	RAPID APPLICATION DEVELOPMENT	LTP	Cr
		310	4

## **OBJECTIVE**

The course on RAD focuses on building applications within a very short time period. After successful completion of this course the students will be able to obtain a firm foundation on RAD concepts and methodologies and acquire sufficient working knowledge in RAD tools.

## **PRE-REQUISITES**

Knowledge of programming in C, C++, JAVA

## **COURSE OUTCOMES**

**CO1:** Understand various strategies for rapid application development (e.g. Agile, Extreme, Joint, Lean, Scrum, Spiral)

**CO2:** Understand the role of scalability in RAD and available solutions (e.g. cloud solutions from Google, Amazon, Microsoft)

**CO3:** Understand the advantages and disadvantages of using platform independent data storage techniques (e.g. Java Data Objects, Java Persistence API) and platform dependent data storage techniques (e.g. Google Datastore)

**CO4:** Proficiently use the programming language within the strategies for RAD.

**CO5:** Create a rapid prototype for the user interface of an IT application

## **UNIT I**

**RAPID APPLICATION DEVELOPMENT:** Introduction to RAD Definition, history; importance strategy; constraints; advantages and disadvantages; customer oriented development;

## **UNIT II**

Different RAD tools: Introduction of different RAD tools; open source versus licensed software builder; Easy Eclipse, Visual studio .net and its. Major components: Various applications build using visual studio; familiarity with IDE visual studio.

## **UNIT III**

.NET Overview & Getting Started with Visual Basic.Net:: Building blocks of .NET framework, Architecture; .NET programming languages; MSIL, Metadata, Common Language Runtime, CTS,CLS, Managed code; what's new in VB.Net, building VB.Net Application, Operators, Methods, Properties, Conditionals and Loops.

#### **UNIT IV**

Implementing OOP using Visual Basic.Net: Structures, Object-Oriented features, Procedures, function & Collection, Exception Handling, SDI and MDI Application.

#### **UNIT V**

WindowsFormControls & Data Access: Form, Button, Textbox, Label, Linkbutton, Checkbox, Listbox, Combobox, Radiobutton, Calender, DateTimePicker

**DialogBox:** Font, Color, Openfile, Savefile, Print; Contextmenu, PictureBox, Progressbar, Richtextbox; OLE, DAO Introduction to data access with ADO.NET; components of ADO.NET. Two Applications in VB.NET

#### **TEXT BOOK**

The Complete Reference Visual Basic .NET, Tata McGRAW- Hill Edition, Jeffrey R. Shapiro

#### **REFERENCE BOOKS**

Visual Basic.Net Programming, Black Book, Dreamtech Press.



<b>BCS-252</b>	<b>COMPUTER GRAPHICS LAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0 0 2</b>	<b>1</b>

### **COURSE OUTCOMES**

**CO1:** Understand the basic concepts of computer graphics.

**CO2:** Design scan conversion problems using C++ programming.

**CO3:** Apply clipping and filling techniques for modifying an object.

**CO4:** Understand the concepts of different type of geometric transformation of objects in 2D and 3D.

**CO5:** Understand the practical implementation of modeling, rendering, viewing of objects in 2D.

### **LIST OF EXPERIMENTS**

1. Write a program for 2D line as raster graphics display using Bresenham line drawing algorithm
2. Write a program for 2D line drawing as raster graphics display using DDA line drawing algorithm
3. Write a program for circle drawing as raster graphics display using mid point circle drawing algorithm
4. Write a program for circle drawing as raster graphics display using Bresenham's circle drawing algorithm
5. Write a program for Transformation operations using
  - a) 2D Translation Transformation
  - b) 2D Scaling Transformation
  - c) 2D Rotation Transformation
6. Write a program for line clipping
7. Write a program for polygon clipping
8. Write a program for display 3D object as 2D raster graphics display using perspective transformation
9. Write a program for rotation for 3D object about arbitrary axis
10. Write a program to create screen saver using graphics Commands

### **REFERENCE BOOKS**

1. Pradeep Bhatia, "Computer Graphics", 2<sup>nd</sup> Edition, I K Publishers, 2009.
2. Computer Graphics by Donald Hearn and M. Pauline Baker, 2nd Edition, Prentice Hall of India, 1999
3. Procedural Elements for Computer Graphics, David F. Rogers, Tata McGraw Hill, Second Edition, 2001

<b>BCS-254</b>	<b>LINUX &amp; SHELL PROGRAMMING LAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0 0 2</b>	<b>1</b>

## **COURSE OUTCOMES**

**CO1** Demonstrate the installation process of various operating systems.

**CO2** Implement virtualization by installing Virtual Machine software

**CO3** Apply UNIX/LINUX operating system commands.

**CO4** Understand different UNIX/LINUX shell scripts and execute various shell programs

## **LIST OF EXPERIMENTS**

1. Study of Linux Operating System
2. Internal System commands and Control Structure in Linux OS
3. Processes and Threads in Linux Operating System
4. Systems and Function Calls in Linux OS
5. Writing of shell scripts in Linux OS
6. AWK Programming in Linux OS
7. Write a shell script to generate a multiplication table.
8. Write a shell script that copies multiple files to a directory.
9. Write a shell script that counts that number of lines and words present in a given file.
10. Write a shell script that displays the list of all files in the given directory.
11. Write a shell script (small calculator) that adds, subtracts, multiplies and divides the given two integers.
12. There are two division options: one returns the quotient and the other returns remainder. The script requires 3 arguments: The operation to be used and two integer numbers. The options are add (-a), subtract (-s), multiply (-m), quotient (-c) and remainder (-r).
13. Write a shell script to reverse the rows and columns of a matrix.

## **REFERENCE BOOK**

Rochkind, M.J., "Advanced Unix Programming", 2<sup>nd</sup> Edition, Pearson Education.

<b>BCS-256</b>	<b>RAPID APPLICATION DEVELOPMENT LAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0 0 2</b>	<b>1</b>

### **COURSE OUTCOMES**

**CO1:** Implementation of the platform independent data storage techniques (e.g. Java Data Objects, Java Persistence API) and platform dependent data storage techniques (e.g. Google Datastore)

**CO2:** Proficiently use the programming language within the strategies for RAD.

**CO3:** Create a rapid prototype for the user interface of an IT application

### **LIST OF EXPERIMENTS/EXERCISES**

1. Program to create 3 forms and perform the following tasks:
  - a. Calculate addition of 5 numbers
  - b. Calculate factorial of a number
  - c. Find whether a given number is even or odd
2. Program to create an employee registration form.
3. Program to create a form and perform the following array operations:
  - a. Sorting of elements of array
  - b. Searching a number in an array
  - c. Merging of 2 given arrays
4. Program to show error handling mechanism in vb.
5. Program to show common dialog control and menu based operations.
6. Program to show functioning of a calculator.
7. Program to create an employee registration form and save the employee records in random files.
8. Program to show employee record and department records using ado control.
9. Program to create data reports in VB
  - a. Data report showing employee records stored in database.
  - b. Data report showing employee list according to their respective departments.
10. Program to create a data report in VB to show the salary details of the employee whose id is entered

**Syllabus**  
**of**  
**BSC**  
**3<sup>rd</sup> year**  
**5<sup>th</sup> Semester**



BCS-301	SOFTWARE ENGINEERING PRINCIPLES	L T P	Cr
		4-0-0	4

### OBJECTIVE

To provide basic knowledge of properties of software and its development processes, software quality, CASE tools, etc.

### PRE-REQUISITES

Knowledge of computer programming, principles of management

### COURSE OUTCOMES

The students undergoing this course will be able:

- CO1:** To learn the basic concepts of software engineering
- CO2:** To know about the requirements and process to engineer the software
- CO3:** To learn how to design a software & what are its strategies
- CO4:** To aware about the coding, testing & maintenance of software
- CO5:** To know about different metrics used for software evaluation

### UNIT I

**INTRODUCTION:** Introduction to Software Engineering, Definition of Software Engineering, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Applications, Software Myths. Software Development Life Cycle Model: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.

### UNIT II

**SOFTWARE REQUIREMENT SPECIFICATIONS:** Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, Control Flow Model, SRS Document, IEEE Standards for SRS, Data Dictionary.

### UNIT III

**SOFTWARE DESIGN:** Basic Concept of Software Design, Architectural Design, Low Level Design: Modularization, Design Structure Charts, Flow Charts, Coupling and Cohesion Measures, Design Strategies: Function Oriented Design, Top-Down and Bottom-Up Design.

### UNIT IV

**CODING & SOFTWARE TESTING & MAINTENANCE:** Top-Down and Bottom –Up programming, structured programming, Code Inspection, Compliance with Design and Coding Standards. Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Top-Down and Bottom-Up Testing Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Alpha and Beta Testing of Products. Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering.

### UNIT V

**SOFTWARE MEASUREMENT & MATRICES:** Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO), Resource Allocation Models, Software Risk Analysis and Management. , Quality Assurance, Quality Control, Software Quality Attributes, Software Quality Assurance (SQA): Verification and Validation

### **TEXT BOOK:**

Pressman Roger S., “Software Engineering – A Practitioner’s Approach”, 6<sup>th</sup> Edition, McGraw Hill, 2004.

### **REFERENCE BOOKS**

1. Aggarwal KK, Singh, Yogesh, “Software Engineering”, New Age International, 2000.
2. Jalote Pankaj,”An Integrated Approach to Software Engineering”, 3<sup>rd</sup> edition, Narosa, 2005.
3. Sommerville Ian, Pearson Education, “Software Engineering”, 5<sup>th</sup> edition, Addison Wesley, 1999.





BCS-303	PROGRAMMING USING C#	L-T-P	Cr
		4-0--0	4

### **OBJECTIVE**

To equip students with C# programming Concepts

### **COURSE OUTCOMES**

The students undergoing this course will be able:

CO1: To know the basics of Dot net programming language

CO2: To aware about different programming languages via differences & comparison

CO3: To learn about the basics of C# programming

CO4: To know about the different statements and controls in C# programming

CO5: To learn about different database connectivity's like ADO etc.

### **UNIT I**

**PHILOSOPHY OF .NET AND ITS MAJOR COMPONENTS:** Origin of .NET technology; .NET platform; benefits and limitations of .NET; building blocks of .NET framework; .NET programming languages; .NET types and namespaces; Understanding CLR, CTS and CLS; developing C# Applications using Visual Studio .Net

### **UNIT II**

**EVOLUTION OF C#:** comparison among C++; Java and C#; benefits of C#; object-oriented programming using C#

### **UNIT III**

**C# PROGRAMMING:** introduction to C#; creating a C# program; types in C#; classes; inheritance and polymorphism; methods; statements and control; arrays and strings; interfaces; abstract and base classes.

### **UNIT IV**

**STATEMENTS AND CONTROL:** properties and indexers; delegates and their usefulness; I/O in C#; exception and error handling in C#.

### **UNIT V**

**ADO .NET AND ASP.NET:** comparison of ADO and ADO. NET; introduction to data access with ADO.NET components of ADO.NET; Comparison of ASP and ASP .NET; features of ASP .NET; features provided by ASP .NET; web forms and their components.

### **TEXT BOOK:**

Balaguruswamy, E, "Programming in C#", Tata McGraw Hill

### **REFERENCE BOOKS**

1. Jain, V K, "The Complete Guide to C# Programming", IDG Books India.
2. Pappas & Murray, "C# Essentials", Prentice Hall of India
3. Gunnerson Eric, "A programmer's Introduction to C#", IDG Books
4. Wakefield, "C# and .NET Web Developers Guide", IDG Books India.

BCS-305	ARTIFICIAL INTELLIGENCE	L T P	Cr
		4-0-0	4

### **OBJECTIVE**

To introduce about artificial intelligence approaches to problem solving, various issues involved and application areas

### **PRE-REQUISITES**

Knowledge of neural networks, data structures

### **COURSE OUTCOMES**

The students undergoing this course will be able:

**CO1:** To know the basics of artificial intelligence

**CO2:** To learn the different searching techniques of artificial intelligence

**CO3:** To learn about the representation of different information to produce a system

**CO4:** To know about different logics used

**CO5:** To aware about the uncertainties

### **UNIT I**

**INTRODUCTION:** Definition of Artificial Intelligence (AI), Evolution of Computing History of AI, data, information and knowledge; AI problems and techniques – AI programming languages; problem space representation with examples, Applications of Artificial Intelligence.

### **UNIT II**

**SEARCH STRATEGIES GAME PLAYING:** Breadth first search; Depth first search; heuristic search techniques: Hill climbing; Best first search; A\* algorithm; AO\* algorithm; Minimax search procedure.

### **UNIT III**

**PRODUCTION SYSTEM & KNOWLEDGE BASE REPRESENTATION:** Production rules, the working memory, Recognize-act cycle, conflict resolution by Meta rules, Architecture of production system. Semantic net, Frames.

### **UNIT IV**

**PROPOSITIONAL LOGIC & PREDICATE LOGIC:** Proposition, tautologies, Theorem proving, forward chaining, backward chaining, method of substitution, Alphabet of first order logic(FOL), predicate, well-formed formula, clause form, algorithm for writing sentence into clause form, Unification of predicates, unification algorithm, resolution Robinson's interface rule.

### **UNIT V**

**REASONING UNDER UNCERTAINTY:** reasoning under uncertainty; non monotonic reasoning; review of probability; Baye's probabilistic interferences and Dempster Shafer theory; Heuristic methods; Fuzzy reasoning.

### **TEXT BOOK**

Elaine Rich and Kevin Knight, "Artificial Intelligence", 3rd Edition, Tata McGraw Hill, 1991

### **REFERENCE BOOKS**

1. Nils J Nilson, "Artificial Intelligence", Harcourt Asia Pvt. Ltd.
2. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall of India, 1998
3. O. W. Patterson, "Introduction to Artificial Intelligence & Expert Systems", Prentice Hall of India
4. Patrick Henry Winston, "Artificial Intelligence", 3rd Edition, Addition Wesley, 1992
5. Programming PROLOG, Clockson & Mellish, Narosa Publications

BCS-307	INTRODUCTION TO E-COMMERCE	L T P	Cr
		3-0-0	3

### **OBJECTIVE**

To give the students knowledge about the e-business and transactions done electronically

### **Pre-Requisites**

Knowledge of internet and web development, data mining, computer networks, software engineering.

### **COURSE OUTCOMES**

The students undergoing this course will be able:

CO1: To learn the basic concepts of e-commerce

CO2: To aware about the infrastructure required in e-commerce

CO3: To know about the payment system operated electronically

CO4: To aware about e-business and its related aspects

CO5: To learn about e-commerce

### **UNIT I**

**INTRODUCTION TO E-COMMERCE:** Benefits; impact of e-commerce; classification of e-commerce; application of e-commerce technology; business models; framework of e-commerce.; business to business; business to customer; customer to customer; advantages and disadvantages of e-commerce; electronic commerce environment and opportunities: back ground – the electronic commerce environment – electronic market place technologies.

### **UNIT II**

**NETWORK INFRASTRUCTURE OF E-COMMERCE:** Network infrastructure to e-commerce & internet; LAN; Ethernet ( IEEE 802.3); WAN; internet; TCP/IP reference model; domain names; internet industry structure; FTP applications; protocols required for ecommerce; HTTP; CGI 3; firewalls; securing web service; secure payment system transaction security (SET); cryptology; digital signatures

### **UNIT III**

**ELECTRONIC PAYMENT SYSTEM & EDI:** Introduction to electronic cash and electronic payment schemes – internet monitory payment; different models; framework; prepaid and post-paid payment model and security requirements – payment and purchase order process – online electronic cash. Search tools: directories; search engines; Meta search engines. EDI & E-content: Business Trade Cycle; EDI; EDI Fact, Electronic content.

### **UNIT IV**

**E-BUSSINESS:** Business requirements – concepts; payment processing. launching your e business- marketing an e-business; public relations; consumer communication; news groups & forums; exchanging links; web rings; e-business back end systems; business record maintenance; back up procedures and disaster recovery plans.

### **UNIT V**

**M-COMMERCE, ADVERTISING & CRM:** Introduction to mobile commerce; framework; applications; design methodology and advantages; future trends in m-commerce. Supply chain management in e-commerce. Internet Advertising; Models of Internet advertising; sponsoring content; Corporate Website; Weaknesses in Internet advertising; web auctions. E-retailing; Role of retailing in E-commerce; E-marketing and advertising. CRM in e-commerce.

**CASE STUDY:** Discussion on a corporate web site, E-commerce legal issues & cyber laws.

**TEXT BOOK**

Chaffey, Dave, “E-business and E-commerce Management”, Pearson Education

**REFERENCE BOOKS**

1. Kalakota, Ravi, Whinston Andrew B . , “E-Commerce-A Manager’s guide”, Addison Wesley.
2. David Whetley; E-commerce concepts.



<b>BCS-353</b>	<b>PROGRAMMING USING C# LAB</b>	<b>L-T-P</b>	<b>Cr</b>
		<b>0-0-2</b>	<b>2</b>

### **OBJECTIVE**

To train the students with C# programming Concepts using implementation

### **COURSE OUTCOMES**

The students undergoing this course will be able:

CO1: To know the basics of Dot net programming language

CO2: To implement operators used in C# Programming

CO3: To learn about constructors to be implemented

CO4: To know about the different functions & file operations in C# programming

CO5: To learn about different database connectivity's like ADO etc.

### **LIST OF EXPERIMENTS**

1. Write a program in C# illustrating the use of sequence, conditional and iteration construct.
2. Write a program in C# illustrating various operators like logical, arithmetical, relational, etc.
3. Write a program in C# illustrating overloading of various operators.
4. Write a program in C# illustrating use of friend, inline and Static Member functions, default arguments.
5. Write a program in C# illustrating use of destructor and various types of constructor.
6. Write a program in C# illustrating various forms of inheritance.
7. Write a program in C# illustrating use of virtual functions, Virtual base class, delegates.
8. Write a program in C# illustrating file operations.
9. Write a program in C# illustrating simple web applications using ASP.net
10. Write a program in C# illustrating use of Active X Controls.

<b>BCS-355</b>	<b>ARTIFICIAL INTELLIGENCE LAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0-0-2</b>	<b>1</b>

### **OBJECTIVE**

To introduce about artificial intelligence approaches to problem solving, various issues involved and application areas

### **PRE-REQUISITES**

Knowledge of neural networks, data structures

### **COURSE OUTCOMES**

The students undergoing this course will be able:

- CO1:** To know the basics of prolog programming language
- CO2:** To learn the different iterative structures using prolog
- CO3:** To learn about different problems and solution by prolog
- CO4:** To know about different searching methods
- CO5:** To aware about the natural language processing

### **LIST OF EXPERIMENTS**

1. Study of Prolog programming language
2. Write programs to use iterative structures using Prolog (at least 3 programs)
3. Write programs to demonstrate inferencing/ deductive logic using Prolog (at least 3 programs)
4. Write a program to solve 8 queens problem using Prolog.
5. Solve any problem using depth first search using Prolog.
6. Solve any problem using best first search using Prolog.
7. Solve 8-puzzle problem using best first search using Prolog
8. Solve Robot (traversal) problem using means End Analysis using Prolog.
9. Solve traveling salesman problem using Prolog.
10. Write program to exhibit the ability of building an Expert System using Prolog
11. Study the properties and issues of Natural Language Processing
12. Study the grammar mapping issues in language translation from English to Hindi and vice versa

### **REFERENCE BOOKS**

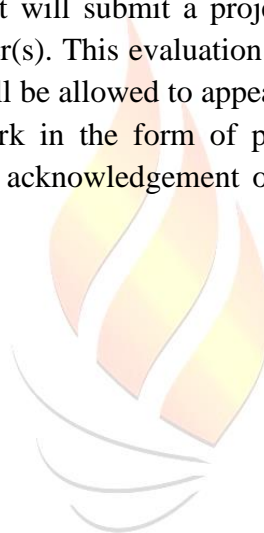
1. Clockson & Mellish, "Programming PROLOG", Narosa Publications, 3rd Edition, 2002.
2. Winston Patrick Henry, "Artificial Intelligence", 3rd Edition, Addition Wesley, 1992

BCS-371	MINOR PROJECT	L T P	Cr
		0-0-8	4

### **OBJECTIVE**

The student shall be capable of identifying a problem related to the program of study and carry out wholesome research on it leading to findings which will facilitate development of a new/improved product, process for the benefit of the society.

The projects should be socially relevant and research oriented ones. Student is expected to do an individual project or in group of 3 members. The project work is carried out in two phases – Minor Project in V semester and Major Project in VI semester. Major project of the project work shall be in continuation of Minor Project only. At the completion of a project the student will submit a project report, which will be evaluated (end semester assessment) by duly appointed examiner(s). This evaluation will be based on the project report and a viva voce examination on the project. Student will be allowed to appear in the final viva voce examination only if he / she has submitted his / her project work in the form of paper for presentation / publication in a conference / journal and produced the proof of acknowledgement of receipt of paper from the organizers / publishers.



<b>BCS-309A</b>	<b>NETWORK SECURITY MANAGEMENT</b>	<b>L T P</b>	<b>Cr</b>
		<b>3 0 0</b>	<b>3</b>

## OBJECTIVE

The main objective behind this course is to learn about the various network attacks and preventing attacks. This course is designed to cover Application security, Network security, Web security etc.

## PRE-REQUISITES

Knowledge of data communications and computer networks, computer programming, data structures, mathematics, telecom network. Knowledge of digital signal processing is desirable

1. **Attacks on Computers & Computer Security:** Introduction; The need of Security ; Security Approaches; Principal of Security; Types of Attacks
2. **Cryptography :** Introduction; Plain Text & Cipher Text; Substitution Techniques; Transposition Techniques; Types of Cryptography; Steganography; Symmetric Key Algorithm: Algorithm Types and Modes, DES; Asymmetric Key Algorithm: RSA, Digital Signatures
3. **Digital Certificates and Public Key Infrastructure:** Digital Certificates ; Private Key Management; The PKIX Model ; Public Key Cryptography Standards; Creating Digital certificates using Java
4. **Internet Security Protocols:** Introduction; Secure Socket Layer(SSL); Secure Electronic Transaction(SET); Electronic Money; Email security; Wireless application protocol (WAP);Security in GSM: Security in 3G
5. **User Authentication and Kerberos:** Introduction ; Authentication Basics; Passwords; Authentication Tokens; Certificate Based Authentication; Biometric Authentication; Kerberos

## REFERENCE BOOKS

1. Atuk Kahate , “Cryptography and Network Security”, 3<sup>rd</sup> Edition, Tata Mcgraw Hill

## REFERENCE BOOKS

2. Stallings William, “Cryptography and Network Security”, 4th Edition, Prentice-Hall, Englewood Cliffs, 2006
3. Mani Subramanian, “Network Management Principles & Practices”, Addison Wesley, 1999
4. Kauffman C., Perlman R. and Spenser M., “Network Security”, 2nd Edition, Prentice Hall, 2002.
5. Menezes Alfred, van Oorschot Paul, and Vanstone Scott, “Handbook of Applied Cryptography”, CRC Press, NY, 2004.
6. Bellovin S. and Chesvick W., “Internet Security and Firewalls”, 2nd Edition, Addison Wesley, 1998. Schneier Bruce, “Applied Cryptography”, Wiley Student Edition, 2nd Edition



<b>BCA-309B</b>	<b>COMPILER DESIGN</b>	<b>L T P</b>	<b>Cr</b>
		<b>3-0-0</b>	<b>3</b>

### **COURSE OBJECTIVE**

To make the student to understand the process involved in a compiler, create an overall view of various types of translators, linkers, loaders, and phases of a compiler, understand what is syntax analysis, various types of parsers especially the top down approach, awareness among students the various types of bottom up parsers, understand the syntax analysis and, intermediate code generation, type checking, the role of symbol table and its organization, Code generation, machine independent code optimization and instruction scheduling.

### **PRE-REQUISITES**

Knowledge of automata theory, context free languages, computer architecture, data structures and simple graph algorithms, logic or algebra.

### **COURSE OUTCOMES**

The students undergoing this course will be able to:

- CO1:** Student will be familiar with the front-end as well as back-end stages of compiler design and Design Lexical analyzer for given language using LEX tools
- CO2:** understand the differences between constructing lexers/parsers by hand versus using automated generators
- CO3:** Hands-on experience with generating intermediate representations, which in turn will let them appreciate the importance of designing simpler languages
- CO4:** To appreciate the nuances of analyzing and transforming programs for performance
- CO5:** Experience of working with relatively large programming environments, which will also inculcate a sense of good software design

### **UNIT I**

**INTRODUCTION TO COMPILING & LEXICAL ANALYSIS:** Introduction of Compiler, Major data Structure in compiler, BOOT Strapping & Porting, Compiler structure: analysis-synthesis model of compilation, various phases of a compiler, Lexical analysis: Input buffering, Specification & Recognition of Tokens, LEX

## **UNIT II**

**SYNTAX ANALYSIS:** Basic Parsing Techniques: Parsers, Shift reduce parsing, Operator precedence parsing, top down parsing, predictive parsers Automatic Construction of efficient Parsers: LR parsers, the canonical Collection of LR(0) items, constructing SLR parsing tables, constructing Canonical LR parsing tables, Constructing LALR parsing tables, using ambiguous grammars, an automatic parser generator, implementation of LR parsing tables.

## **UNIT III**

**SYNTAX DIRECTED TRANSLATION:** Syntax directed definitions: Construction of Syntax trees, Bottom up evaluation of S-attributed definition, L-attribute definition, Top down translation, Bottom Up evaluation of inherited attributes Recursive Evaluation, Analysis of Syntax directed definition.

## **UNIT IV**

**CODE GENERATION:** Intermediate code generation: Declarations, Assignment statements, Boolean expressions, Case statements, Procedure calls Code Generation: Issues in the design of code generator, Basic block and flow graphs, register allocation and assignment, DAG representation of basic blocks, peephole optimization, generating code from DAG.


## **UNIT V**

**CODE OPTIMIZATION:** Introduction to Code optimization: sources of optimization of basic blocks, loops in flow graphs, dead code elimination, loop optimization, Introduction to global data flow analysis, Code Improving transformations, Data flow analysis of structure flow graph Symbolic debugging of optimized code.

## **REFERENCES:**

1. V. Aho, R. Sethi, and J. D. Ullman. Compilers: Principles, Techniques and Tools, Pearson Education
2. Raghavan, Compiler Design, TMH Pub.
3. Loudon. Compiler Construction: Principles and Practice, Cengage Learning
4. Holub. Compiler Design in C , Prentice-Hall Inc., 1993.
5. Mak, writing compiler & Interpreters, Willey Pub.

Syllabus  
of  
BSC  
3<sup>rd</sup> Year  
6<sup>th</sup> Semester



<b>BCS-302</b>	<b>SOFTWARE PROJECT MANAGEMENT</b>	<b>L T P</b>	<b>Cr</b>
		<b>3 0 0</b>	<b>3</b>

## **OBJECTIVE**

To provide the foundation required for becoming a good software project manager by means of planning, evaluation and estimation, risk management, allocation and monitoring of resources, controlling software quality PRE-REQUISITES Knowledge of software engineering and the basic principles of management

## **COURSE OUTCOMES**

The students after undergoing this course will be able to:

**CO1:** Identify the different project contexts and suggest an appropriate management strategy.

**CO2:** Practice the role of professional ethics in successful software development.

**CO3:** Identify and describe the key phases of project management.

**CO4:** Determine an appropriate project management approach through an evaluation of the business context and scope of the project.

**CO5:** Methods to Maintain the Software Quality Assurance

## **UNIT I**

**INTRODUCTION:** Definition of a Software Project (SP), SP vs. other types of projects activities covered by SPM; categorizing SPs; project as a system; management control, requirement specification; information and control in organization

**STEPWISE PROJECT PLANNING:** Introduction, selecting a project; identifying project scope and objectives; identifying project infrastructure, analyzing project characteristics; identifying project products and activities; estimate efforts each activity; identifying activity risk; allocate resources; review/ publicize plan

## **UNIT II**

**PROJECT EVALUATION AND ESTIMATION:** Cost benefit analysis; cash flow forecasting; cost benefit evaluation techniques; risk evaluation; Selection of an appropriate project report; Choosing technologies, choice of process model, structured methods: rapid application development, water fall, V-process-, spiral-models; Prototyping;

**ACTIVITY PLANNING :** Objectives of activity planning; project schedule; projects and activities; sequencing and scheduling activities, network planning model; representation of lagged activities; adding the time dimension, backward and forward pass; identifying critical path; activity throat, shortening project; precedence networks;

### **UNIT III**

**RESOURCE ALLOCATION AND MONITORING THE CONTROL:** Introduction, the nature of resources, identifying resource requirements; scheduling resources creating critical paths; counting the cost; being specific; publishing the resource schedule; cost schedules, the scheduling sequence; Monitoring the control: Introduction, creating the frame work, collecting the data, visualizing progress, cost monitoring, earned value, prioritizing monitoring, getting the project back to target, change control

### **UNIT IV**

**MANAGING CONTRACTS AND PEOPLE:**, Introduction, types of contract, stages in contract, placement, typical terms of a contract, contract management, acceptance, Managing people and organizing terms: Introduction, understanding behavior, organizational behavior: a back ground, selecting the right person for the job, instruction in the best methods, motivation, working in groups, becoming a team, decision making, leadership, organizational structures, conclusion, further exercises

### **UNIT V**

**SOFTWARE QUALITY:** Introduction; the place of software quality in project planning; the importance of software quality; defining software quality, ISO 9126; Practical software quality measures; product versus process quality management; 46 external standards; techniques to help enhance software quality; Study of any software project management software: viz Project 2005 or equivalent.

### **TEXT BOOKS**

Bob Hughes and Mike Cotterell, —Software Project Manag em entl, 2nd Edition, Tata McGraw Hill, 1999

### **REFERENCE BOOKS**

1. Futrell, —Software Quality & Project Manag em entl, Pearson Education, 2002.
2. Jalote Pankaj, Software Project Management, Pearson Education, 2002.
3. Gopaldaswam y Ramesh, —Managing Global Software Projectsll, Tata McGraw Hill, 2001
4. Pressman Roger S., —Software Engineering – A Practitioner’s Approachll, 5th Edition, McGraw Hill, 2001
5. Walker Royce, —Software Project Manag em entl, Addison Wesley, 1998
6. Maylor, —Project Manag em entl, Third Edition, 2003.
7. Demarco Tom, —Controlling Software Project Management and Measurem entl, Prentice Hall, 1982 8.  
Glib Tom and Susannah Finzi, —Principles of Software Engineering Managementll, Addison Wesley, 1998.

<b>BCS-304</b>	<b>NEURAL NETWORK</b>	<b>L T P</b>	<b>Cr</b>
		<b>4 0 0</b>	<b>4</b>

### **OBJECTIVE**

To study various algorithms and their implementation in real life and in different domains.

### **PRE-REQUISITES**

Knowledge of Applied Mathematics and basic calculation techniques of matrices.

### **COURSE OUTCOMES**

The students undergoing this course will be able to:

**CO1:** Model Neuron and Neural Network, and to analyze ANN learning, and its applications

**CO2:** Perform Pattern Recognition, Linear classification.

**CO3:** Develop different single layer/multiple layer Perception learning algorithms

**CO4:** Design of another class of layered networks using deep learning principles.

### **UNIT I**

Overview of biological neurons: Structure of biological neurons relevant to ANNs.

### **UNIT II**

Fundamental concepts of Artificial Neural Networks: Models of ANNs; Feed forward & feedback networks; learning rules; Hebbian learning rule, perception learning rule, delta learning rule, Widrow-Hoff learning rule, correction learning rule, Winner take all learning rule, etc.

### **UNIT III**

Single layer Perception Classifier: Classification model, Features & Decision regions; training & classification using discrete perception, algorithm, single layer continuous perception networks for linearly separable classifications.

### **UNIT IV**

Multi-layer Feed forward Networks: linearly non-separable pattern classification, Delta learning rule for multi-perceptron layer, generalized delta learning rule, Error back propagation training, learning factors, Examples.

### **UNIT V**

Single layer feedback Networks: Basic Concepts, Hopfield networks, Training & Examples, associative memories

### **TEXT BOOKS**

Introduction to artificial neural systems by Jacek M. Zurada, 1994, Jaico Publ.House.

### **REFERENCE BOOKS**

1. Neural Networks :A Comprehensive formulation , Simon Haykin, 1998, AW
2. Neural Networks , Kosko, 1992, PHI.
3. Neural Network Fundamentals N.K. Bose , P. Liang, 2002, T.M.H

<b>BCS-310</b>	<b>CRYPTOGRAPHY AND DATA COMPRESSION</b>	<b>L T P</b>	<b>Cr</b>
		<b>3 0 0</b>	<b>3</b>

### **OBJECTIVE**

The course will provide a down-to-earth overview of cryptographic techniques applicable in an IT environment, and outline the constraints and limitations of realistic secure systems. A running theme is the tradeoff between usability and security of a system. Also covered are a number of compression techniques - data compression and data encryption are, in some respects, closely related. A working knowledge of C is assumed and essential.

### **COURSE OUTCOMES**

**CO1:** Understand and analyze public-key cryptography, RSA and other public-key cryptosystems

**CO2:** Analyze and design hash and MAC algorithms, and digital signatures.

**CO3:** Design network application security schemes, such as PGP, S/ MIME, IPSec, SSL, TLS, HTTPS, SSH, etc.

**CO4:** Understand key management and distribution schemes and design User Authentication Protocol

**CO5:** Know about Intruders and Intruder Detection mechanisms, Types of Malicious software, Firewall Characteristics, Types of Firewalls, Firewall Location and Configurations.

### **UNIT I**

**COMPRESSION:** Packing; Huffman coding; run length encoding; Lempel-Ziv-Welch; Phil Katz's PKZIP; Delta modulation; JPEG.

### **UNIT II**

**ERROR DETECTION AND CORRECTION:** Parity; 1, 2, n-dimensions, Hamming codes; p-out-of-q codes

### **UNIT III**

**CRYPTOGRAPHY:** Vocabulary; history, steganography – visual, textual; cipher hiding; false errors; public key cryptography - authentication, signatures, deniability

### **UNIT IV**

**MATHEMATICS:** Information; confusion; diffusion; modular arithmetic; inverses; Fermat's little theorem, Chinese remainder theorem; factoring; prime numbers; discrete logarithms

### **UNIT V**

**ALGORITHMS:** DES; AES (Rijndael); IDEA; one time pad; secret sharing and splitting; RSA; elliptic curves; modes; random numbers

## REFERENCE BOOKS

1. IEEE, “Integration of Data Compression and Cryptography: Another Way to Increase the Information Security”, IEEE Computer Society
2. Schneier B., “Applied Cryptography: Protocols, Algorithms and Source Code in C”, 2nd edition, Wiley, 1996.
3. Desai Suhag, “Security in Computing”, Pearson Education
4. Trappe W. and Washington L., “Introduction to Cryptography”, 2nd edition, Pearson Education, 2006





BCS-308	PYTHON PROGRAMMING	L-T-P	Cr
		3-0-0	3

## OBJECTIVE

To build programming logic and thereby developing skills in problem solving using Python programming language; To be able to do testing and debugging of code written in Python Emphasize the concepts and constructs rather than on language features.

## COURSE OUTCOMES

The students undergoing this course will be able to:

**CO1:** To learn and understand Python programming basics and paradigm.

**CO2:** To learn and understand python looping, control statements and string manipulations.

**CO3:** Students should be made familiar with the concepts of GUI controls and designing GUI applications.

**CO4:** To learn and know the concepts of Structure and Functions.

**CO5:** To learn and know the concepts of file handling, exception handling and database connectivity.

## UNIT I

**PLANNING THE COMPUTER PROGRAM AND PROBLEM-SOLVING TECHNIQUES:** Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation. Flow charting, decision table, algorithms, structured programming concepts, Programming methodologies viz. Top-down and bottom-up programming.

## UNIT II

**OVERVIEW OF PROGRAMMING & INTRODUCTION TO PYTHON:** Structure of a Python Program, Elements of Python. Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators (Arithmetic Operator, Relation a l operator, Logical or Boolean operator, Assignment, Operator, Ternary operator ,Bit wise operator, Increment or Decrement operator).

## UNIT III

**CREATING PYTHON PROGRAMS:** Input and Output Statements, Control Statements(Looping- while Loop, for Loop ,Loop Control ,Conditional Statement-if...else, Difference between break ,continue and pass).

## UNIT IV

**STRUCTURES& FUNCTIONS:** Numbers, Strings, Lists, Tuples, Dictionary, Date & Time, Modules, Defining Functions, Exit function, default arguments.

## UNIT V

**CLASSES, OBJECT-ORIENTED PROGRAMMING AND EXCEPTION:** Abstract Data Types and Classes, Inheritance, Encapsulation and information hiding, Handling exceptions

### **TEXT BOOKS**

John V Guttag. "Introduction to Computation and Programming Using Python", Prentice Hall of India

### **REFERENCE BOOKS**

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Python Tutorial/Documentation [www.python.org](http://www.python.org)2010
3. Allen Downey, Jeffrey Elkner, Chris Meyers ,How to think like a computer scientist :Learning with Python, Freely available online.2012
4. <http://docs.python.org/3/tutorial/index.html>
5. <http://interactivepython.org/courselib/static/pythonds>
6. <http://www.ibiblio.org/g2swap/byteofpython/read/>



<b>BCS-310A</b>	<b>ELECTIVE – II (BIG DATA ANALYSIS)</b>	<b>L T P</b>	<b>Cr</b>
		<b>3 0 0</b>	<b>3</b>

## **OBJECTIVE**

The basics of Hadoop, the basics of Analytics – Concepts, Data preparation – merging, managing missing numbers sampling, Data visualization and Basic statistics.

## **COURSE OUTCOMES**

The students undergoing this course will be able to:

**CO1:** Identify Big Data and its Business Implications.

**CO2:** List the components of Hadoop and Hadoop Eco-System

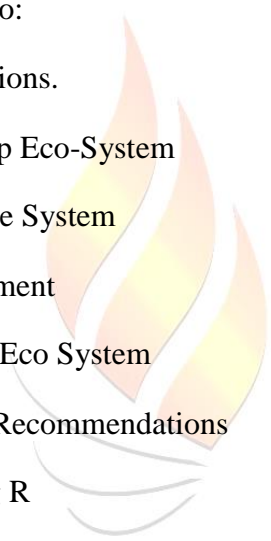
**CO3:** Access and Process Data on Distributed File System

**CO4:** Manage Job Execution in Hadoop Environment

**CO5:** Develop Big Data Solutions using Hadoop Eco System

**CO6:** Analyze Infosphere Big Insights Big Data Recommendations

**CO7:** Apply Machine Learning Techniques using R



## **UNIT I**

**INTRODUCTION TO BIG DATA :** Introduction – distributed file system – Big Data and its importance, Four Vs, Drivers for Big data, Big data analytics, Big data applications. Algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce, Challenges for processing big data, Using big data in businesses.

## **UNIT II**

**INTRODUCTION TO HADOOP:** Introduction to Hadoop, why we use Hadoop, History of Hadoop, Use cases of Hadoop, Big Data – Apache Hadoop – Moving Data in and out of Hadoop – Understanding inputs and outputs of MapReduce - Data Serialization.

## **UNIT III**

**HADOOP ARCHITECTURE:** Hadoop Architecture, Hadoop Storage: HDFS, Common Hadoop Shell commands , Anatomy of File Write and Read., Name Node, Secondary Name Node, and Data Node, Hadoop MapReduce paradigm, Map and Reduce tasks, Job, Task trackers - Cluster Setup – SSH & Hadoop Configuration – HDFS Administering –Monitoring & Maintenance.

## **UNIT IV**

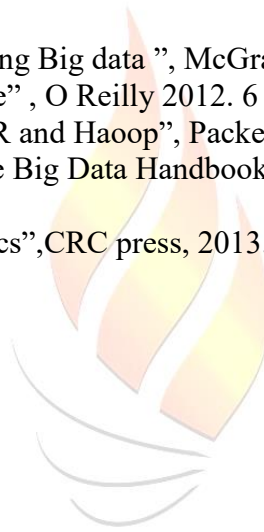
**HADOOP ECOSYSTEM AND YARN:** Learning MapReduce concepts and framework, Testing and Debugging Map Reduce Applications, Background of YARN; Hadoop YARN architecture; advantages of YARN, working with YARN, backward compatibility with YARN, YARN Commands, log management etc.

## **UNIT V**

**HIVE AND HBASE:** Introduction to Hive and HBASE, HIVE: Architecture, Managing tables, data types, schemas, partitions, HBASE: Architecture, Schema design; Advance Indexing - PIG, Zookeeper - how it helps in monitoring a cluster, HBase uses Zookeeper and how to Build Applications with Zookeeper., HBASE commands, HIVE Vs RDMS, HBASE Vs RDMS.

## **REFERENCES**

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley, ISBN: 9788126551071, 2015.
2. Chris Eaton, Dirk deroos et al. , “Understanding Big data ”, McGraw Hill, 2012.
3. Tom White, “HADOOP: The definitive Guide” , O Reilly 2012. 6 IT2015 SRM(E&T)
4. Vignesh Prajapati, “Big Data Analytics with R and Haoop”, Packet Publishing 2013.
5. Tom Plunkett, Brian Macdonald et al, “Oracle Big Data Handbook”, Oracle Press, 2014.
6. <http://www.bigdatauniversity.com/>
7. Jy Liebowitz, “Big Data and Business analytics”,CRC press, 2013.



<b>BCS-354</b>	<b>NEURAL NETWORK LAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0 0 2</b>	<b>1</b>

### **COURSE OUTCOMES**

**CO1:** Implement the Neuron and Neural Network, and to analyze ANN learning, and its applications

**CO2:** Implement and Perform Pattern Recognition, Linear classification.

**CO3:** Develop different single layer/multiple layer Perception learning algorithms

### **LIST OF EXPERIMENTS**

1. Study of NN toolbox
2. Study of MATLAB functions
3. To perform basic matrix operations.
4. To plot the following
  - (a) A straight line
  - (b) A sine curve.
5. To generate a few activation functions that is used in neural networks in MATLAB.
6. To plot hard limit transfer function.
7. To generate XOR function using McCulloch Pitts Neural Network in MATLAB.
8. To generate AND-NOT function using McCulloch Pitts Neural Network in MATLAB.
9. To use Hebbian Network to classify 2-Dimensional input pattern.
10. Write a MATLAB program for perceptron net for and function with bipolar input and targets.

### **REFERENCE BOOKS**

Haykin Simon, —Neural Networks: A Comprehensive Formulation, Addison Wesley

<b>BCS-358</b>	<b>PYTHON PROGRAMMING LAB</b>	<b>L T P</b>	<b>Cr</b>
		<b>0 0 2</b>	<b>1</b>

## **COURSE OUTCOMES**

**CO1** Define and demonstrate the use of built-in data structures “lists” and “dictionary”.

**CO2** Design and implement a program to solve a real-world problem

**CO3** Design and implement GUI application and how to handle exceptions and files.

**CO4** Make database connectivity in python programming language.

## **LIST OF PROGRAMS**

### **Program 1: Programs using if else structure**

- a) Find the Largest Among Three Numbers
- b) Python Program to Check Leap Year
- c) Python Program to Take in the Marks of 5 Subjects and Display the Grade
- d) Python Program to Check if a Date is Valid and Print next date

### **Program 2: Programs using for and while loop**

- a) Python Program to check whether given number is Prime Number or not
- b) Python program to compute the GCD of two numbers
- c) Python Program to Find the Sum of Digits in a Number
- d) Python Program to convert binary number to decimal number
- e) Python Program to Display Fibonacci sequence Using Recursion

### **Program 3: Program using List and String data structure**

- a) Write Python Program to input a list of integers, (1) display the no of elements in the list (2) display minimum and maximum element in the list (3) display sum of square of all the element in the list (4) (5) add a new element at end and display the list (6) add a new element at given index and display list (7) display the occurrence of given element in the list (8) remove the given element in the list (9) add element from a new list from given list (10) sort the given list & reverse the given list (11) also perform slicing, concatenation and multiplication operation
- b) A fruit seller sells different type of fruits. Type of fruits and corresponding rates are stored in two different lists. Customer can order any type of fruit (one or more type) in any quantity. If total bill of customer is greater than 500, customer is given 10% discount. If any of the fruits required by the customer is not available in the store, then consider the bill amount to be -1. Write a Python program to calculate and display the bill amount
- c) Write a Python program to display all the permutations of given string (don't use python permutation function)
- d) Accept two strings 'string1' and 'string2' as input from the user. Generate a resultant string-1, such that it is a concatenated string of all upper case alphabets from both the strings in the order they appear.

Generate a resultant string-2 that contain character which are in both string1 and 2 Print the actual resultant string-1 and resultant string-2

---

#### **Program 4: Programs using concept of sets, tuple & dictionary**

- Write a Python program that take a string as input and store the character and occurrence of each character in a dictionary. Create two lists from dictionary first having each character in sorted order of their frequency and second having corresponding frequency.
- A furniture seller sells different type of furniture, Type of Furniture and rates are stored in a dictionary. Customer can order any type of furniture (one or more type) in any quantity. If total bill of customer is greater than 10,000, customer is given 5% discount. 8% GST is charged on total bill. If any of the furniture required by the customer is not available in the store, then consider the bill amount to be -1. Write a Python program to calculate and display the bill amount
- Consider a scenario from Lingayas Vidyapeeth. Given below are two Sets representing the names of students enrolled for a particular course: `java_course = {"Anmol", "Rahul", "Priyanka", "Pratik"}`  
`python_course = {"Rahul", "Ram", "Nazim", "Vishal"}` Write a Python program to list the number of students enrolled for:1)Python course2)Java course only3)Python course only4)Both Java and Python courses5)Either Java or Python courses but not both6)Either Java or Python
- Students name and their corresponding marks are stored in a dictionary. Write a Python program to perform following (1) Display name and marks of each student (2) Display the names of top two scorer (3) display the class average for this course (4) check if the marks for given student is stored in dictionary or not, if not add the name and marks in the dictionary else display his/her marks (5) delete the name and marks of a given student in the dictionary (6) add name and marks from another dictionary and display combined dictionary

#### **Program 5: Program using Function in Python:**

- Write Python functions using the concept of Keyword & default arguments and write a program to use them
- Write python functions to use the concept of variable length argument & global variable. Write a program to use these functions

#### **Program 6: Program using concept of Class, object, class variable, class method:**

- Create a class Account with name, account no and balance as attribute and `no_of_accounts` as class variable. Account no should be generated automatically (starting from 1) using the class variable `no_of_account`. Add the methods for displaying the account information, depositing given amount, withdrawing given amount and initializer method to initialize the object. Create objects of Account class and call different method to test the class
- Create a class Employee with name, empid, salary as attribute and `no_of_employee` and `annual_incr` (% annual increment) as class variable. empid should be generated automatically (starting from 1) using the class variable, `no_of_employee`. Add the instance methods for displaying the employee information, annually increasing the salary with help of class variable `annual_incr`, class method to change the value of `annual_incr` and initializer method to initialize the object. Create objects of employee class and call different method to test the class (program using class method)
- Write a Program to showing the use of built in class attributes (`__doc__`, `__dict__`, `__name__`, `__module__`, `__bases__`) and special methods(`__del__()`, `__str__()`) and built in function `isinstance()`

#### **Program 7: Program using the concept of Inheritance**

---

- Create a class Polygon to represent a polygon having no of sides and a list having magnitude of each side as attribute. Add the inputSides() to input sides and displaySides() to display sides as methods. Derive a class Triangle from Polygon and add an additional method displayArea() to display area. Create object of Triangle and call different methods to test the class
- Create a class Person having name, age, as attributes, `__init__()` method to initialize the object and display() to display person information. Derive a class Student from Person having roll no, University

name, branch as additional attributes and `__init__()`, `display()` to display student information and `change_Branch()` method. Create object of Student type and call different methods to test the class.

- c) Write a program to show the concept of multiple inheritance in python

---

### Program 8: Program using the concept of Polymorphism, Operator Overloading

- a) In a retail outlet there are two modes of bill Payment (1) Cash : Calculation includes VAT(10%) Total Amount = Purchase amount + VAT (2) Credit card: Calculation includes processing charge and VAT Total Amount = Purchase amount + VAT (10%) + Processing charge (2%) The act of bill payment is same but the formula used for calculation of total amount differs as per the mode of payment. Can the Payment maker simply call a method and that method dynamically selects the formula for the total amount? Demonstrate this Polymorphic behaviour with code.
- b) Write a program to create a class to represent length in feet and inch. Overload the “+” operator to add the two object of length type.
- c) Write a program to overload comparison operator in python

---

### Program 9: Program on file handling in Python

- a) Write a python program to write few lines on a file, read it back and create a dictionary having each word in file as keys in dictionary and occurrence of these word as values and print the dictionary.
- b) A file student.txt store student information. Information about each student is written on separate line in the form: roll-no student-name (student-name may consist of any number of words).Write a Python program that takes student roll no as input and print the student name. If roll no is not present in the file it display : “roll no not present in the file”
- c) Write a python program to read a file that contains email ids on the separate lines in the form: “personname@companyname.com. Create a new file that contain only company names, read the new file to print the company name

---

### Program 10: Program on Exception handling

- a) Write a function divide (arg1, arg2) to divide arg1 by arg2. Use the exception handling mechanism to handle all type of possible exceptions that may occur. Take the value of arg1 and arg2( of any type) from user as input and call the function divide to print the result of division or suitable message if any type of exception occurs( use also else and finally block)
- b) Write a program to open a file in read only mode read data from file and then try to write data on file. Use the exception handling mechanism to handle all type of possible exception
- c) Write a Python program that takes email id, mobile number and age as inputs from user. Validate each and raise user defined exceptions accordingly

Note:-

Email id: there must be only one @ and At least one “.”

Mobile number must be 10 digits

Age must be a positive number less than 101



BCS-372	MAJOR PROJECT	L-T-P	CR
		0-0-10	5

### **OBJECTIVES**

1. Identify and discuss the role and importance of research in the emerging Technology and Engineering
2. Ability to synthesize knowledge and skills previously gained and applied to an in-depth study and execution of new technical problem
3. Capable to select from different methodologies, methods and forms of analysis to produce a suitable research design, and justify their design.
4. Ability to present the findings of their technical solution in a written report.
5. Presenting the work in International/ National conference or reputed journals

### **COURSE OUTCOMES**

After undergoing this course, the students will be able to:

**CO1:**Develop aptitude for research and independent learning.

**CO2:**Demonstrate the ability to carry out literature survey and select unresolved problems in the domain of the selected project topic

**CO3:**Gain the expertise to use new tools and techniques for the design and development.

**CO4:**Acquire the knowledge and awareness to carry out cost-effective and environment friendly designs.

**CO5:**Develop the ability to write good technical report, to make oral presentation of the work, and to publish the work in reputed conferences/journals.

The Major project topic should be selected / chosen to ensure the satisfaction of the urgent need to establish a direct link between education, national development and productivity and thus reduce the gap between the world of work and the world of study.

The Major research Project should have the following

- Relevance to social needs of society
- Relevance to value addition to existing facilities in the institute
- Relevance to industry need
- Problems of national importance
- Research and development in various domain
- Literature survey Problem Definition
- Motivation for study and Objectives
- Preliminary design / feasibility / modular approaches
- Implementation and Verification
- Report and presentation
- Experimental verification / Proof of concept
- Design, fabrication, testing of Communication System.

<b>BCS-310B</b>	<b>ELECTIVE- II (Cloud Computing)</b>	<b>L T P</b>	<b>Cr</b>
		<b>3 0 0</b>	<b>3</b>

**Course Objective:** This course gives students an insight into the basics of cloud computing along with virtualization, cloud computing is one of the fastest growing domain from a while now. It will provide the students basic understanding about cloud and virtualization along with it how one can migrate over it.

### **Unit-1: Overview of Delivery models in Cloud Computing**

Cloud Computing Platform Overview, Why Cloud Computing?; Evolution of Cloud Computing; What is Cloud Computing?; Essential characteristics of Cloud Computing; Types of Cloud, Cloud Computing Advantages; Illustration of the benefits of cloud computing ; Cloud Computing Challenges; Illustration of cloud computing challenges; Cloud Computing Service models; Cloud Computing Deployment models; Cloud Service and Deployment models; Cloud adoption considerations; Cloud adoption; Cloud History – Internet technologies (SOA, Web Services, Web 2.0, mashups), Distributed computing – Utility and Grid Computing; Hardware – VMWare ESXi, Xen, KVM; Virtual Appliances and the open Virtualization format; System Management; Anatomy of Cloud; Benefits of Cloud; Cloud Transformation roadmap; cloud delivery models and their advantages; Cloud computing architecture.

### **Unit-2: IaaS, PaaS and SaaS**

Introduction to Infrastructure as a Service delivery model; characteristics of IaaS, Architecture, examples of IaaS; Applicability of IaaS in the industry, Comparing ISPs and IaaS, Motivations for renting the infrastructure; IaaS; Case studies; IaaS enabling Technology; Trusted cloud; Introduction to Platform as a Service delivery model, characteristics of PaaS, patterns, architecture and examples of PaaS; Applicability of PaaS in the industry ; integrated Lifecycle Platform; Anchored Lifecycle platform; Enabling Technologies as a Platform; PaaS – best option or not. Introduction to Software as a Service delivery model, characteristics of SaaS, SaaS Origin; Evolvement of SaaS – Salseforce.com’s approach; SaaS Economics and Ecosystem; Types of SaaS Platforms; Architecture; SaaS – Providers; Collaboration as a Service; Enabling and Management tools as a Service; Applicability of SaaS in the industry.

### **Unit-3: Cloud Computing Reference Architecture (CCRA)**

Introduction to Cloud computing reference architecture (CCRA); benefits of CCRA, Architecture overview – The conceptual Reference Model; Cloud Consumer; Cloud provider; Cloud Auditor; Cloud carrier; Scope of control between Provider and Consumer; CCRA : Architectural Components – Service deployment , Service Orchestration, Cloud Service Management, Security; Cloud Taxonomy; IBM’s Cloud Computing Reference Architecture(CCRA 2.0) – Introduction, roles, Architectural elements; CCRA evolution; Examples of Cloud Services; versions and application of CCRA for developing clouds.

### **Unit-4: Private, Public Cloud Deployment Models**

What is a Private Cloud?, Illustration of Private Cloud, Advantages of Private Cloud, Limitations of Private Cloud; Service Management, Journey into Private Cloud; Planning and Strategy, Standardization, Virtualization, Automation, Cloud, Case study – VMware vCloud; Case Study – IBM SmartCloud Entry, Private cloud; What is a Public Cloud?, Illustration of Public Cloud, Why Public Cloud, Advantages of Public Cloud, Limitations of Public Cloud; Low degree of security and control, Lack of control on infrastructure, configuration, Network latency and accessibility concerns; Highest long term cost, Public v/s Private, Journey into Public Cloud, Revisit the idea of adopting public cloud, Cloud vendor selection; Migrating to Cloud,

Cloud vendor selection, SLA – Service Level Agreements, Credits/Compensation terms, Credit process, Disaster recovery plan, Exclusions, Security and Privacy, Periodic upgrade and maintenance; Data location and Jurisdiction, Pricing and Measurability, Interoperability and Lock-in, Exit process/Termination policies, Proven track record, Public cloud vendors,

### **Unit-5: Hybrid Cloud**

Case studies. What is a Hybrid Cloud?, Why Hybrid Cloud, Illustration of Hybrid Cloud, Advantages of Hybrid Cloud, Challenges of Hybrid Cloud, Develop and manage hybrid workloads, Developing applications for hybrid cloud; Develop applications using PaaS, Managing hybrid workloads; Journey into Hybrid Cloud, Step 1: Asses current IT infrastructure and business, Step 2: Explore cloud computing, Step 3: Create cloud deployment strategy plan, Step 4: hybrid cloud implementation.

#### **Text & References: Text:**

- Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter, TATA McGraw- Hill , New Delhi – 2010
- Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008

#### **References:**

- Cloud computing for dummies- Judith Hurwitz , Robin Bloor , Marcia Kaufman ,Fern Halper, Wiley Publishing, Inc, 2010
- Cloud Computing (Principles and Paradigms), Edited by Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 2011

\*\*\*\*\*