

**KALASALINGAM UNIVERSITY**  
**(KALASALINGAM ACADEMY OF RESEARCH AND EDUCATION)**  
**(Under Section 3 of the UGC Act 1956)**  
**Anand Nagar, Krishnankoil-626126**  
**Srivilliputtur(via); Virudhunagar(Dt.), Tamil Nadu, INDIA**  
**(www.kalasalingam.ac.in)**



**B.Sc(Computer Science)**  
**(Bachelor of Science)**  
**CURRICULUM AND SYLLABUS - 2014**



**KALASALINGAM UNIVERSITY**  
**(KALASALINGAM ACADEMY OF RESEARCH AND EDUCATION)**  
**(Under Section 3 of the UGC Act 1956)**  
**Anand Nagar, Krishnankoil-626126**  
**Srivilliputtur(via); Virudhunagar(Dt.), Tamil Nadu, INDIA**  
([www.kalasalingam.ac.in](http://www.kalasalingam.ac.in))

## **DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

### **VISION**

**To become a Center of excellence offering quality education and innovation in Computer Science and Information Technology.**

### **MISSION**

- **To prepare the students to excel in the field of Computer Science and IT industry**
- **To prepare the students to pursue higher studies and develop sustainable innovative solutions for the society.**



**KALASALINGAM UNIVERSITY**  
**(KALASALINGAM ACADEMY OF RESEARCH AND EDUCATION)**  
**(Under Section 3 of the UGC Act 1956)**  
**Anand Nagar, Krishnankoil-626126**  
**Srivilliputtur(via); Virudhunagar(Dt.), Tamil Nadu, INDIA**  
 (www.kalasalingam.ac.in)  
 DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY  
**B.Sc (Computer Science)**

### **Programme Educational Objectives (PEOs)**

PEO 1	Graduates of the programme will be employed in the field computer Science.
PEO 2	Graduates of the programme will pursue higher studies.
PEO3	Graduates of the programme will apply new technologies in Computer Science to serve the needs of industry, and society.

### **Programme Outcomes (Pos)**

- Po1 : Ability to apply knowledge in mathematics, science fundamentals to solve problems.
- Po2 : Understand the basic concepts of system software, hardware and evolution of computer graphics.
- Po3 : Ability to use a range of programming languages and tools to develop computer programs that are effective to solve the problems.
- Po4 : Understand the basic concept of computer architectures, including computer hardware and networking.
- Po5 : Design, and analyze precise specifications of algorithms, procedures, and interaction behavior.
- Po6 : Ability to communicate effectively in both verbal and written form in industry and society.
- Po7 : Ability to work in teams to build software systems.
- Po8 : Apply the technologies in various fields of Computer Science, including Mobile applications, Web site development and management, databases, and computer networks.
- Po9 : Ability to select appropriate techniques to tackle and solve problems in the discipline of information security management.

**Department of Computer Applications**  
**B.Sc(Computer Science) Programme**

**I Semester**

Subject Code	Subjects	L	T	P	C
	<b>PART - I</b>				
BAE111	Tamil	3	0	0	3
	<b>PART – II</b>				
BAE112	Prose , Poetry, Fiction, Grammar, Vocabulary, Spoken English and Composition	3	0	0	3
	<b>PART-III</b>				
BCS101	Discrete Mathematics	3	1	0	4
BCS102	Digital Principles and Computer Architecture	3	1	0	4
BCS103	Programming in C	3	0	0	3
BCS181	Digital Electronics Laboratory	0	0	3	2
BCS182	Programming in C Laboratory	0	0	3	2
	<b>TOTAL</b>	<b>15</b>	<b>2</b>	<b>6</b>	<b>21</b>

**II Semester**

Subject Code	Subjects	L	T	P	C
	<b>PART - I</b>				
BAE121	Tamil	3	0	0	3
	<b>PART – II</b>				
BAE122	Prose , Poetry, Fiction, Grammar, Vocabulary, Spoken English and Composition	3	0	0	3
	<b>PART-III</b>				
BCS201	Probability and Statistics	3	1	0	4
BCS202	Data Structures	3	1	0	4
BCS203	Object Oriented Programming in C++	3	0	0	3
BCS281	Data Structures Laboratory	0	0	3	2
BCS282	Object Oriented Programming Laboratory	0	0	3	2
	<b>TOTAL</b>	<b>15</b>	<b>2</b>	<b>6</b>	<b>21</b>

**SECOND YEAR  
III Semester**

<b>Subject Code</b>	<b>Subjects</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
BCS301	Graph Theory	3	0	0	3
BCS302	Java Programming	3	1	0	4
BCS303	Operating Systems	3	0	0	3
BCS304	Microprocessor and its Applications	3	1	0	4
CHY102	Environmental Science	2	0	0	2
BCS306	Computer Algorithms	3	0	0	3
BCS381	Java Programming Laboratory	0	0	3	2
BCS382	Operating Systems Laboratory	0	0	3	2
	<b>TOTAL</b>	<b>17</b>	<b>2</b>	<b>6</b>	<b>23</b>

**IV Semester**

<b>Subject Code</b>	<b>Subjects</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
BCS401	Operations Research	3	1	0	4
BCS402	Web Technologies	3	0	0	3
BCS403	Database Management Systems	3	1	0	4
BCS404	Computer Networks	3	1	0	4
BCS405	PC Hardware and Trouble Shooting	3	0	0	3
BCS481	Web Technologies Laboratory	0	0	3	2
BCS482	RDBMS Laboratory	0	0	3	2
BCS483	Communicative English Laboratory	0	0	2	1
	<b>TOTAL</b>	<b>15</b>	<b>3</b>	<b>8</b>	<b>23</b>

**THIRD YEAR****V Semester**

<b>Subject Code</b>	<b>Subjects</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
BCS501	System Software	3	0	0	3
BCS502	Software Engineering	3	1	0	4
BCS503	Open Source Technologies	3	1	0	4
BCS504	Computer Graphics	3	0	0	3
BCS51*	Elective-I	3	0	0	3
BCS581	Open Source Technologies Laboratory	0	0	3	2
BCS582	Graphics Laboratory	0	0	3	2
	<b>TOTAL</b>	<b>15</b>	<b>2</b>	<b>6</b>	<b>21</b>

**VI Semester**

<b>Subject Code</b>	<b>Subjects</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
BCS601	Visual Programming	3	1	0	4
BCS602	Software Testing	3	0	0	3
BCS61*	Elective-II	3	0	0	3
BCS681	Visual Programming Laboratory	0	0	3	2
BCS682	Software Testing Laboratory	0	0	3	2
BCS698	Project Work	0	0	8	6
	<b>TOTAL</b>	<b>9</b>	<b>1</b>	<b>14</b>	<b>20</b>

**ELECTIVES****Fifth Semester**

<b>Subject Code</b>	<b>Subjects</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
BCS511	Advanced Databases	3	-	-	3
BCS512	Mobile Computing	3	-	-	3
BCS513	E-Commerce	3	-	-	3

**Sixth Semester**

<b>Subject Code</b>	<b>Subjects</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
BCS611	Cloud Computing	3	-	-	3
BCS612	Data Mining and Data Warehousing	3	-	-	3
BCS613	Information Security	3	-	-	3

**Consolidated CGPA Credits**

<b>Semester</b>	<b>Credits</b>
I – Semester	21
II – Semester	21
III – Semester	23
IV – Semester	23
V – Semester	21
VI – Semester	20
<b>Total Credits</b>	<b>129</b>

**FIRST YEAR  
SEMESTER-I**

<b>BAE111</b>	<b>TAMIL</b>	L	T	P	C
		3	0	0	3

**ghlj;jpl;l;jpd;; Fwpf;Nfhs;:**

- khzth;fsplk; thrpf;Fk; gof;fj;ij Vw;gLj;Jtjd; %yk; rpwe;j khzth;fis cUthf;fyhk;
- nkhopj;jpwidf; fw;Fk; NghJ khzth; ve;jr; nrhy;iy vq;F gad;gLj;j Ntz;Lk; vd;w Mw;wYk; Vw;gLfpwJ
- ,yf;fzk; fw;gpf;Fk; nghOJ gioa kuGfis kPwhky; Gjpa cj;jpfisg; gad;gLj;jp Mh;tj;ijj; J}z;lyhk;
- ,yf;fpa czh;it kdjpy; tpijg;gJ fUj;Jg; ghpkhw;wj;jpw;Fr; rhjfkfh mikfpwJ.

**ghlj;jpl;l;jpd;; ntspg;ghL:**

CO 1. jkpo; GJf;ftpij gilg;Gfis mwpKfg;gLj;Jjy;	(Ghpjy;)
CO 2. GJf;ftpij gilg;ghsh;fs; mth;jk; gilg;Gfs;	(Ghpjy;)
CO 3. GJf;ftpij; \$Wfs; ghlg;gFjp ftpijfs;	(nghUj;jpg; ghh;j;jy;)
CO 4. jkpo;r; rpWfij tuyhW	(Ghpjy;)
CO 5. gilg;G cj;jpfs; Fwpg;gplj;jf;f gilg;Gfs;	(Ghpjy;)
CO6. ghlg;gFjp rpWfijfSk; ,yf;fzq;fSk;	(epidt+l;ly;)
CO7. mbg;gil kw;Wk; ghlg;gFjp ,yf;fzk; (cUthf;Fjy;)	

SEMESTER I	Part I	Paper – I	Code :BAE 111	Periods 6 / Credits: 3
<b>Kjy; gUtk; : ,f;fhy ftpijAk; ehlfKk;</b>				
myF – I : kuGf; ftpij – ghujpahh;> ghujpjhrd;> Rujh				
myF – II : ehl;Lg;Gwghly;fs;				
myF – III : GJf;ftpij (20-Mk; E}w;whz;L etPd ftpQh;fs;)				



ehlfk; - itifapy; nts;sk; tUk;.
myF – IV : ,yf;fzk;
myF –V : ,yf;fpa tuyhWk; gad;ghl;Lj; jkpOk;
kh.tujuh\$d; jkpH; ,yf;fpa tuyhW rhfpj;a mflhkp 2001.
nrJgjp. itfiuapy; bts;sk;tUk;. ghit gjpg;gfk; 2007.

<b>BAE112</b>	<b>PROSE, POETRY, FICTION, GRAMMAR, VOCABULARY, SPOKEN ENGLISH AND COMPOSITION</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- To help the learners to have good critical ability
- To help the learner to prepare short speeches and deliver them effectively
- To help the learners to actively participate in story writing
- To help the learners to have fluency in English

**COURSE OUTCOMES**

On successful completion of this course, the students will be able to

1. Know more about Mahatma Gandhi, Mother Teresa, Martin Luther King
2. Describe Daffodils, beauty of Byron's Maid, painful account of apple- pickers
3. Apply the concept of the stories to the present cult
4. Apply the concept of the stories to the present cult
5. Understand the basic Grammar, and Spoken English
6. Ability to write composition, letter and vocabulary
7. Gain vocabulary through reading
8. Acquire fluency in English language

**PROSE**

V.S.Srinivasa Sastri -- Mahatma Gandhi  
 John Frazer -- Mother Terasa  
 R.N.Roy -- Martin Luther King

**POETRY**

Wordsworth -- "Daffodils"  
 Byron -- "She Walks in Beauty"  
 Robert Frost -- "After Apple-picking"

## **FICTION: SHORT STORIES**

O. Henry -- “Gift of the Magi”  
Chekhov -- “The Bet”  
Rabindranath Tagore -- “The Postmaster

## **GRAMMAR AND SPOKEN ENGLISH**

- Noun
- Pronouns
- Possessive forms
- Articles & Nouns
- Demonstratives
- Some, any, no etc.

(Response in one or two sentences)

Greeting a person; Introducing oneself; Inviting somebody to attend a function; Expressing inability to attend; Requesting; Seeking permission; Refusing permission; Asking for advice; Remembering something; Expressing sympathy; Reminding others; Congratulating; Complaining; Apologizing; Making suggestions; Warning; Asking for information/direction; Expressing annoyance; Encouraging others; Expressing possibility/impossibility; Starting a conversation with a stranger; Ending a conversation; Asking for someone’s opinion; Expressing happiness; Expressing something unpleasant; Expressing gratitude.

## **COMPOSITION AND VOCABULARY**

### **Composition**

- (1) Reading comprehension: 8 questions testing skills of locating direct information, associative comprehension, overall understanding, drawing inferences, evaluative comprehension and aspects of grammar and vocabulary.
- (2) Arranging jumbled sentences in a chronological order or a coherent paragraph.
- (3) Letter Writing (Personal letters).

### **Vocabulary**

One Word Substitutes:

alimony, amateur, amnesty, anaesthesia, anarchist, anatomy, anonymous, archive, atheist, autobiography, cannibal, carcinogen, cardiologist, carnivorous, centenarian, contemporary, connoisseur, cosmopolitan, crew, detective, (21 – 40) emigrant, epitaph, extempore, fauna, feminist, fleet, flora, forgery, gymnasium, gynaecologist, herbivorous, hypocrisy, incorrigible, kleptomania, lexicographer, manuscript, mercenary, misanthrope, mortuary, novice, (41 – 60) obituary, omniscient, ophthalmologist, optimist, omnipotent, orphan, panacea, parasite, pedestrian,

pessimist, philanthropy philatelist, polygamy, posthumous, post-mortem, secular, somnambulist, theology, unanimous, utopia.

### BOOKS PRESCRIBED:

Jegadisan, S. **Portraits in Prose**. Orient Black Swan, Chennai: 2009.

Sadanand Kamalesh. & Punitha, Susheela. **Spoken English: A Foundation Course**. Part 2  
Orient Black Swan, New Delhi, 2011

<b>BCS101</b>	<b>DISCRETE MATHEMATICS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

### COURSE OBJECTIVES

1. To familiarize the students with the concept and techniques of propositional logic and equivalences and their application to logic theory.
2. To study the Graph and Graph modules.
3. To grasp the concepts in lattices as algebraic system.

### COURSE OUTCOME

Upon successful completion of this course, students will be able to:

1. Explain and illustrate the concept of proposition disjunction, conjunction, and conditional statement and their use in solving problems.
2. Explain and illustrate the concept of mathematical induction and its use in solving problems.
3. Demonstrate the concept of graphs and graph models, marginal and conditional probability distribution involving two random variables.
4. Explain and illustrate algebraic systems, semigroups, monoids and homomorphism.
5. Understand the concept of lattices and special type of Boolean algebra.

### UNIT- I

Propositional Logic – Propositional equivalences-Predicates and quantifiers-Nested Quantifiers-Rules of inference-introduction to Proofs-Proof Methods and strategy

### UNIT- II

Mathematical inductions-Strong induction and well ordering-.The basics of counting-The pigeonhole principle –Permutations and combinations-Recurrence relations Solving Linear recurrence relations-generating functions-inclusion and exclusion and applications.

### UNIT -III

Graphs and graph models-Graph terminology and special types of graphs-Representing graphs and graph isomorphism -connectivity-Euler and Hamilton paths

#### UNIT -IV

Algebraic systems-Semi groups and monoids-Groups-Subgroups and Homomorphisms Cosets and Lagrange's theorem- Ring & Fields (Definitions and examples)

#### UNIT -V

Partial ordering-Posets-Lattices as Posets- Properties of lattices-Lattices as Algebraic systems –Sub lattices –direct product and Homomorphism-Some Special lattices Boolean Algebra

#### TEXT BOOKS:

- 1.Kenneth H.Rosen, "Discrete Mathematics and its Applications",Special Indian edition,Tata McGraw-Hill Pub. Co. Ltd., New Delhi,2011.
2. Tremblay J.P and Manohar R, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw–Hill Pub. Co. Ltd, New Delhi, 30<sup>th</sup> edition 2007.

#### REFERENCES:

1. Ralph. P. Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", Fourth Edition, Pearson Education Asia, Delhi, 2009.
2. Thomas Koshy, "Discrete Mathematics with Applications", Elsevier Publications, 2006.

BCS102	DIGITAL PRINCIPLES AND COMPUTER ARCHITECTURE	L	T	P	C
		3	1	0	4

#### COURSE OBJECTIVES

- This course is designed to provide a comprehensive introduction to digital logic design leading to the ability to understand number system representations, binary codes, binary arithmetic and Boolean algebra, its axioms and theorems, and its relevance to digital logic design.
- To Analyze and design simple systems composed of programmable logic, such as ROMs and PLAs.
- Aiming at conducting Tutorial, seminars and remedial classes.

#### COURSE OUTCOME

Upon successful completion of this course, students will be able to

1. Demonstrate knowledge of binary number theory, Boolean algebra and binary codes.
2. Analyze and design combinational systems using standard gates and minimization methods (such as Karnaugh maps).
3. Analyze and design combinational systems composed of standard combinational modules, such as multiplexers flip-flops, demultiplexer and decoders.
4. Demonstrate knowledge of simple synchronous sequential systems.
5. Analyze and design sequential systems composed of standard sequential modules, such as counters and registers.
6. Analyze and design simple systems composed of programmable logic, such as ROMs and PLAs.
7. Perform basic arithmetic operations with signed integers represented in binary.

### **UNIT-I**

Digital Computers and digital systems – Binary numbers – Number base conversion – Octal and Hexadecimal numbers – Complements – Binary codes.

### **UNIT-II**

Basic definitions of Boolean functions – Digital logic gates – Simplification of Boolean functions – Karnaugh map method.

### **UNIT-III**

Design of Half Adder, Full Adder, Half Subtractor , Full Subtractor, Ripple Adders and Subtractors, Ripple Adder/Subtractor Using Ones and Twos Complement Method. Serial Adder , Carry Look Ahead Adder, Design of Decoders, Encoders, Multiplexers, Demultiplexers.-Flip-flops –

### **UNIT-IV**

Processor organization – Organization of a processor - Registers, ALU and *Control* unit,. Data path in a *CPU*, Instruction cycle, Organization of a control unit - Operations of a control unit -.Control organization – Execution of instructions.

### **UNIT V**

Micro computer organization – Instruction and addressing modes – Memory organization Memory Classification-RAM-ROM --PROM-EPROM-EAPROM-Programmable Logic Array – Direct Memory Access.

### **TEXT BOOK**

1. M. Moris Mano, Computer System Architecture, 3<sup>rd</sup> edition, Prentice Hall.

### **REFERENCES**

1. W. Stallings, Computer Organization and Architecture, 6<sup>th</sup> edition, PHI, 2003.
2. C. Hamacher, Z. Vranesic, S.Zaky, Computer Organization, 5<sup>th</sup> edition, Mcgraw Hill, 2002.

<b>BCS103</b>	<b>PROGRAMMING IN C</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

### **COURSE OBJECTIVES**

- This course is designed to provide a comprehensive study of the C programming language. It stresses the strengths of C, which provide students with the means of writing efficient, maintainable, and portable code. The nature of C language is emphasized in the wide variety of examples and applications.

### **COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Understand the basic terminology used in computer programming
2. Write, compile and debug programs in C language.
3. Use different data types in a computer program.
4. Design programs involving decision structures, loops and functions.
5. Explain the difference between call by value and call by reference
6. Understand the dynamics of memory by the use of pointers and Structures.
7. Use different data structures and create/update basic data files.

### **UNIT-I**

History of C - Characteristics of C - C Program Structure - Data Types - Variables and Constants - Operators - Conditional Statements - Looping and Iteration

### **UNIT-II**

Single Dimensional Array - Multi Dimensional Array - Types of functions - Functions and Arrays - String Functions - Recursive Functions

### **UNIT-III**

Basics, Structures and functions - Arrays of structures - Pointers to structures - Self referential structures - Typedef - Union - Bitfields - Enum Data Types

### **UNIT-IV**

Introduction - Pointer Types - Pointers to Strings - Pointers to Array - Pointers to Structure Pointers and Dynamic Allocation of Memory - Pointers to function.

### **UNIT V**

File management and Console input and output – Functions for file management - Standard I/o, Formatted output - Formatted input - File access - Error handling

**TEXT BOOK:**

1. Herbert Schildt, The Complete Reference C, 4th Edition, Tata Mc - Graw Hill, 2000.

**REFERENCES:**

1. Byron C Gottfried, Programming with C, Schaums' outline series, 2nd Edition, Tata Mc - Graw Hill, 2006.
2. Brian Kernighan, W., Dennis Ritchie, M., The C Programming Language, 2nd Edition, Prentice Hall of India Pvt. Ltd., 2005

<b>BCS181</b>	<b>DIGITAL ELECTRONICS LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	2

**COURSE OBJECTIVES**

- To understand and use number system representations, binary codes, binary arithmetic and Boolean algebra and its relevance to digital logic design.

**COURSE OUTCOME**

Upon successful completion of this lab course, students will be able to

1. Represent numerical values in various number systems and perform number conversions between different number systems.
2. Analyze and design digital combinational circuits like decoders, encoders, multiplexers, and de- multiplexers including arithmetic circuits (half adder, full adder, and multiplier).
3. Design and implement binary Adder and subtractor
4. Design and implement 3-bit synchronous counters
5. Implement shift registers using Flip- flops

**PROGRAMS**

1. Design and implementation of Adder and Subtractor using logic gates.
2. Design and implementation of code converters using logic gates
  - (i) BCD to excess-3 code and vice versa
  - (ii) Binary to gray and vice-versa
3. Design and implementation of 2 bit Magnitude Comparator using logic gates 8 Bit Magnitude Comparator using IC 7485.

4. Design and implementation of 4 bit binary Adder/ subtractor
5. Design and implementation of 16 bit odd/even parity checker generator using IC74180.
6. Design and implementation of Multiplexer and De-multiplexer using logic gates and study of IC74150 and IC 74154
7. Design and implementation of encoder and decoder using logic gates and study of IC7445 and IC74147
8. Construction and verification of 4 bit ripple counter and Mod-10 / Mod-12 Ripple counters.
9. Design and implementation of 3-bit synchronous up/down counter
10. Implementation of SISO, SIPO, PISO and PIPO shift registers using Flip- flops

<b>BCS182</b>	<b>PROGRAMMING IN C LABORATORY</b>	L	T	P	C
		0	0	3	2

#### **COURSE OBJECTIVES**

- To write, compile and debug programs in C language.
- To formulate problems and implement algorithms in C.
- To effectively choose programming components that efficiently solve computing problems in real-world.

#### **COURSE OUTCOME**

Upon successful completion of this lab Course, student will be able to

1. Understand the basic concept of C Programming, and its different modules.
2. Acquire knowledge about the basic concept of writing a program.
3. Explain the role of constants, variables, identifiers, operators, type conversion and other building blocks of C Language.
4. Use the conditional expressions and looping statements to solve problems associated with conditions and repetitions.
5. Demonstrate the role of Functions involving the idea of modularity.
6. Understand the concept of Array and pointers dealing with memory management.
7. Use the structures and unions through which derived data types can be formed.
8. Understand the file handling for permanent storage of data.

#### **PROGRAMS**

1. To demonstrate use of data types, simple operators (expressions)
2. To demonstrate decision making statements (if and if-else, nested structures)
3. To demonstrate decision making statements (switch case)
4. To demonstrate use of simple loops



5. To demonstrate use of nested loops
6. To demonstrate menu driven programs and use of standard library functions.
7. To demonstrate writing C programs in modular way ( use of user defined functions)
8. To demonstrate recursive functions.
9. To demonstrate use of arrays (1-d arrays ) and functions
- 10.To demonstrate use of multidimensional array(2-d arrays ) and functions
- 11 To demonstrate use of pointers
1. To demonstrate concept of strings (strings and pointers)
- 13.To demonstrate array of strings.
14. To demonstrate structures (using array and functions)
15. To demonstrate nested structures and Unions
16. To demonstrate file handling (text files

## **SEMESTER-II**

BAE121	TAMIL	L	T	P	C
		3	0	0	3

**ghlj;jpd; Fwvf;Nfhs;:**

- gf;jp ,af;fk; Njhd;wpa tuyhw;iwAk;> mJ jkpofj;jpy; Vw;gLj;jpa jhf;fj;ijAk; khzthplk; GFj;jy;
- ,iwczh;it Cl;b fw;Nghhpd; cs;s;ij cUFkhW vLj;Jiuf;f Ntz;Lk;
- gf;jpg;ghly;fs; thrpf;Fk; NghNj nrhw;nghUspidAk;> fUj;Jf;fisAk; czh;r;rptbtpy; mikj;jy; rpwg;ghFk;
- gpwiu ey;topg;gLj;Jk; Nehf;fpy; gf;jp ,yf;fpak; gad;gLfpwJ

**ghlj;jpd; ntspg;ghL:**

CO 1. gf;jp,yf;fpa Njhw;wg; gpd;Gyk; (Ghpjy;)

CO 2. rkaq;fSk; murpaYk; (Ghpjy;)

CO 3. rpw;wpyf;fpaj;jpd; fhyk; (Ghpjy;)

CO 4. rpw;wpyf;fpa tiffs; (epidT+l;ly;)

CO 5. Gjpdk; - tho;tpay; (nghUj;jpg; ghh;j;jy;)

CO 6. nrhy; tiffs; mwpjy; (Ghpjy;)

CO 7. rkak; tsh;j;j jkpo; (epidT+l;ly;)

SEMESTER II	Part I	Paper – II	Code :BAE 121	Periods 6 / Credits: 3
,uz;lhk; gUtk;				
myF – I	:	,yf;fpa ciueilAk; fij ,yf;fpaKk; (11 rpWfijfs;)		
myF – II	:	Mw;wq;fiu Xuk; (ehty;)		
myF – III	:	,yf;fzk; (nrhy;)		
myF – IV	:	,yf;fpa tuyhW gad;ghl;Lj; jkpo;		
myF –V	:	nra;jp> fl;Liu – vOj;Jgapw;rp		
kh.tujuh\$d;. jkpH; ,yf;fpa tuyhW. rhfpj;a mflhkp 2001.				
uh.nrJgps;is. Mw;w';fiunahuk;. ghit gjpg;gfk; 1977.				

<b>BAE122</b>	<b>PROSE, POETRY, FICTION, GRAMMAR, VOCABULARY, SPOKEN ENGLISH AND COMPOSITION</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**PROSE**

D.H.Lawrence -- The Rocking Horse Winner

Anton Chekov -- The Bet

Somerset Maugham -- Princess September

**POETRY**

William Wordsworth -- "Lucy"

W.B.Yeats -- "The Wild Swans"

D.H.Lawrence -- "The Best of School"

T.S.Eliot -- "To the Indians who Died in Africa"

Philip Larkin -- "That Whitsun"

**FICTION – SHORT STORIES**

William Somerset Maugham -- The Verger

R.K.Narayan -- An Astrologer's Day

Bhabani Bhattacharya -- Glory at Twilight

**GRAMMAR AND SPOKEN ENGLISH****Grammar:**

1. Tenses
2. Active and Passive

**Spoken English** (Responses in one or two sentences)

Making assertions; Describing persons and objects; Taking a vow; Commenting on situations; Exclamations; Giving yourself time to think; Asking for someone's opinion; Asking about preferences; Asking whether someone knows; Saying you know; Checking if

someone has understood; Asking if someone agrees; Asking someone to repeat; Leaving someone for a short time; Asking if someone is able to do something; Saying you are unable to do something; Changing the subject; Avoiding giving an opinion; Ordering food at a restaurant; Talking about weather and season; Asking about daily activities; Making polite requests using, “*Would (Will) you . . . . Please?*”, etc.; Discussing Television Programme.

## COMPOSITION & VOCABULARY

- (1) Letter Writing (Official) and Preparation of *curriculum vitae*.
- (2) Developing hints into a story/general essay; Simple topics to be chosen for essay.
- (3) Note making.

### Vocabulary

Homophones:

beer – bier; berth – birth; deer – dear; desert – dessert; diseased – deceased; feat – feet; heard – herd; cast – caste; check – cheque; draught – draft; hangar – hanger; hear – here; knight – night; loan – lone; mane – main; Meet – meat; meter – metre; pair – pear; peal – peel; personal – personnel; piece – peace -- peas; pray – prey; root – route; sent -- scent – cent; cite -- site – sight; sow – sew; stationary – stationery; steal – steel; story – storey; wait – weight.

### BOOKS PRESCRIBED:

Board of Editors. **Journey through Words**. Orient Longman, Hyderabad: 2007.

Board of Editors. **The Fragrance of Fiction**. Orient Black Swan, Hyderabad: 2011.

<b>BCS201</b>	<b>PROBABILITY AND STATISTICS</b>	L	T	P	C
		3	1	0	4

### COURSE OBJECTIVES

- Apply key concepts of probability, including discrete and continuous random variables, probability distributions, conditioning, independence, expectations, and moments.
- Define and demonstrate the concepts of estimation and properties of estimators and apply the concepts of interval estimation and confidence intervals.

### COURSE OUTCOMES

Students who successfully complete this course should be able to

1. Evaluate the probabilities and conditional probabilities.
2. Evaluate expectations and conditional expectations of random variables.

3. Approximate the distribution of sum of random variables using CLT.
4. Construct point estimators using the method of maximum likelihood.
5. Calculate the number of samples needed to construct confidence levels on the mean and variance of a normal distribution.
6. Use linear regression analysis to develop an empirical model of experimental data.

**UNIT-I**

Introduction - scope and limitations of statistical methods - classification of data - Tabulation of data - Diagrammatic and Graphical representation of data - Graphical determination of percentiles and quartiles.

**UNIT-II**

Measures of location : Arithmetic mean, median, mode, geometric mean and Harmonic mean and their properties.

**UNIT-III**

Measures of dispersion : Range, Quartile deviation, mean deviation, Standard deviation, combined standard deviation, co-efficient of variation.

**UNIT-IV**

Measures of Skew ness Karl Pearson's, Bowley's, kelly's and co-efficient of Skewness and kurtosis based on moments.

**UNIT-V**

Correlation - Karl Pearson - spearman's rank correlation - concurrent deviation methods. Regression Analysis: Simple Regression Equations.

**TEXT BOOK:**

1. Fundamental of Mathematical Statistics - S.C. Gupta & V.K. Kapoor - Sultan Chand

**REFERENCES**

1. Statistical Methods - Snedecor G.W. & Cochran W.G. oxford & +DII
2. Elements of Statistics - Mode . E.B. - Prentice Hall
3. Statistical Methods - Dr. S.P. Gupta - Sultan Chand & Sons

<b>BCS202</b>	<b>DATA STRUCTURES</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

**COURSE OBJECTIVES**

- The objective of this course is to teach students various data structures and to explain them algorithms for performing various operations on these data structures.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Demonstrate familiarity with major algorithms and data structures.
2. Analyze performance of algorithms and choose the appropriate data structure and algorithm design method for a specified application.
3. Determine which algorithm or data structure to use in different scenarios and be familiar with writing recursive methods.
4. Demonstrate understanding of the abstract properties of various data structures such as stacks, queues, lists, trees and graphs and Use various data structures effectively in application programs.
5. Demonstrate understanding of various sorting algorithms, including bubble sort, insertion sort, selection sort, heap sort and quick sort.
6. Understand and apply fundamental algorithmic problems including Tree traversals, Graph traversals, and shortest paths.
7. Gain knowledge about Hashing and Collisions and B- Trees.

### **UNIT – I**

Algorithmic notation – Programming principles – Creating programs- Analyzing programs - array –Representation of Arrays- One dimensional array- Multidimensional array- pointer arrays- Stack and Queue – Fundamentals of stack and Queues – Evaluation of Expressions

### **UNIT II**

Linked List: Single Linked List – Doubly Linked List – Circular Linked List- Application of Linked List- Dynamic Storage Management – Generalized List – Garbage Collection And Compaction

### **UNIT III**

Tree Introduction - Binary Tree Representation – Tree Traversal- Threaded binary Tree – AVL Tree– Graphs – Graphs representation –Graph Traversal - Topological Sorting

### **UNIT –IV**

Bubble sort – Insertion Sort – Selection sort – Radix Sort - Shell Sort– Merge sort – Quick Sort – Heap Sort –Linear Search – Binary Search

### **UNIT V**

Hashing – Types of Hashing – Collision Resolution techniques- B- Tree Representation – B tree Operations- B+ Tree Indexing

### **TEXT BOOKS:**

1. D. Samanta- “Classic Data Structures”- Prentice-Hall of India- Pvt. Ltd.- India 2006.
2. Ellis Horowitz ,Fundamentals of Computer Algorithms, Sartaj Sahni, Rajasekaran, 2<sup>nd</sup> Edition, University Press, 2008.

**REFERENCES :**

1. Robert Kruse- C.L. Tondo and Bruce Leung- “Data Structures and Program Design in C”- Prentice-Hall of India- Pvt. Ltd.- Third- 2006.
2. Jean Paul Tremblay and Paul G. Sorenson- “An Introduction to Data Structures with Applications”- Tata McGraw-Hill- Third Edition- 2006.
3. Mark Allen Weiss-” Data Structures and Algorithm Analysis in C”- Pearson Education- Second edition- 2006

<b>BCS203</b>	<b>OBJECT ORIENTED PROGRAMMING IN C++</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES**

- To get a clear understanding of object-oriented concepts.
- To understand object oriented programming through C++.

**COURSE OUTCOME**

Upon successful completion of this course, Student will be able to

1. Gain the basic knowledge on Object Oriented concepts.
2. Develop applications using Object Oriented Programming Concepts
3. Demonstrate the differences between traditional imperative design and object-oriented design
4. Explain class structures as fundamental, modular building blocks
5. Understand the role of inheritance, polymorphism, dynamic binding and generic structures in building reusable code
6. Write small/medium scale C++ programs with simple graphical user interface
7. Understand the file handling and error handling mechanisms in C++
8. Get knowledge to use strings and Streams in C++
9. Implement features of object oriented programming to solve real world problems

**UNIT-I**

OOP concept, Procedural vs OOP programming, OOP terminology and features, Tokens, Character set, Keywords, Data-types, Data Types declarations, Constants and variables, expressions, Standard Library and header files. Operator and Expressions: Arithmetic Operator, Increment/Decrement Operator, Relational Operator, Logical Operator and conditional operators, library functions, Logical Expressions, C++ shorthand operators- Enumerated Data Types.

**UNIT-II**

Flow of control statements: Selection statements, Iteration statement, Jump statement, Construction of loops and implementation, Classes and Objects: Need for Classes, Declaration of Classes, referencing class Members, Scope of class and its members Nested Classes, Functions in a class: Inline Functions, Constant Member functions, Nesting of

Member Functions, friend function, Memory allocation of objects, Arrays of objects, Static Class Member -Arrays two dimensional and multidimensional arrays, Arrays of Pointers, Pointers and functions.

### UNIT-III

Constructors and Destructor: Declaration, Definition and characteristics, Function Overloading, Inheritance:Need, Different forms of inheritance – Virtual functions, this pointer- Operator Overloading :Overloading Unary Operators, Overloading Binary Operators

### UNIT-IV

File Handling: Classes for file stream operations, opening and closing a file, detecting end of file, file modes, file pointers and their manipulations, sequential input and output operations, random access, file operations error handling, command line argument, Exception Handling- try, catch statements, Multiple catch statements.

### UNIT-V

Strings and Streams: the string class and functions, stream classes, the ios class, ios format flags, ios state , variables ,the istream and ostream classes, unformatted input functions , unformatted output functions, stream manipulators. Templates and Iterators: function templates, class templates, container classes, subclass templates, passing template classes to template parameters, iterator classes.

### TEXT BOOK

1. E Balagurusamy, “Object oriented Programming with C++”, Tata McGraw-Hill Publishing Company, Edition 5 ,June 2011
2. D.Ravichandran, “Programming with C++”, Tata McGraw Hill, Edition 4, 2011.

### REFERENCES

- 1.HM Deitel and PJ Deitel “C++ How to Program”, Prentice Hall ,Seventh Edition, 2010
- 2.Herbert Schildt, “The Complete Reference in C++”, Tata McGraw Hill.Fourth Edition, 2003.
- 3.Y.P.Kanetkar, “Let us C++” , BPB publication,2013.
- 4.Bjarne Stroustrup, “The C++ Programming language”, Addison-Wesley, 2013.

<b>BCS281</b>	<b>DATA STRUCTURES LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	2

### COURSE OBJECTIVE

- To develop skills to design and analyze simple linear and non linear data structures
- To Strengthen the ability to identify and apply the suitable data structure for the given real world problem
- To Gain knowledge in practical applications of data structures

### COURSE OUTCOME



Upon successful completion of this course, students will be able to

1. Design and analyze the time and space efficiency of the data structure
2. Capable to identify the appropriate data structure for given problem
3. Get practical knowledge on the application of data structures
4. Implement linked list data structure to solve various problems.
5. Apply graph and tree traverse technique to various applications.
6. Implement Dijkstra's algorithm, Btrees and hash tables.
7. Understand and apply various data structure such as stacks, queues, trees and graphs to solve various computing problems using C-programming language.

#### PROGRAMS

1. Searching algorithms - sequential- binary
2. Sorting algorithms (any five): Insertion Sort- Selection Sort- Shell Sort- Bubble Sort- Quick Sort- Heap Sort- Merge Sort- and Radix Sort.
3. Sparse Matrix representation and find its transpose.
4. Evaluation of Arithmetic Expression.
5. Queue- circular Queue- Priority Queue- Dequeue.
6. Singly Linked List- Doubly Linked List- Circular Linked List.
7. Tree Traversal Techniques.
8. Graph Traversal Techniques.
9. Dijkstra's algorithm.
10. Use of hash tables.
11. B -Tree Indexing.

<b>BCS282</b>	<b>OBJECT ORIENTED PROGRAMMING IN C++ LABORATORY</b>	L	T	P	C
		0	0	3	2

#### COURSE OBJECTIVE

- To make the student learn an object oriented way of solving problems.
- To make the student to identify and practice the object-oriented programming concepts and techniques.
- To practice the use of C++ classes and class libraries, modify existing C++ classes.
- To develop C++ classes for simple applications

#### COURSE OUTCOME

Upon successful completion of this course, students will be able to

1. Apply object-oriented programming features to program design and implementation
2. Understand object-oriented concepts and how they are supported by C++
3. Understand implementation issues related to object-oriented techniques.
4. Analyze, use, and create functions, classes, to overload operators.
5. Use inheritance and Pointers when creating or using classes and create templates
6. Understand and use Exception handling and file handling mechanism.

7. Write programs that make appropriate use of advanced object-oriented facilities common to many object-oriented languages such as classes, message passing, overloading and inheritance.
8. Design interactive programs with a simple GUI interface using an object-oriented programming language.
9. Choose and apply appropriate advanced object-oriented programming concepts.

#### PROGRAMS

1. Programs using Control Structures
2. Programs using Functions
3. Programs using Arrays
4. Programs using Inline Functions
5. Programs using Classes
6. Programs using Constructors and Destructors
7. Programs using Friend Functions
8. Programs using Operator Overloading
9. Programs using Inheritance
10. Programs using Virtual Functions
11. Programs using Files
12. Programs using Strings

### SEMESTER-III

<b>BCS301</b>	<b>GRAPH THEORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

#### COURSE OBJECTIVES

1. To familiarize the students with the concept of sub graph, walks, path, circuits, connected and disconnected graphs.
2. To understand the properties of tree and cut set and Fundamental circuits.
3. To grasp the concepts in Fundamental circuits and Chromatic number.

#### COURSE OUTCOMES

Upon successful completion of this course, students will be able to

1. Solve some real life problems using techniques of graphs.
2. Understand the properties of tree and cut set and Fundamental circuits
3. Get knowledge about spanning tree of a connected graph
4. Acquire logical skills in the connectivity and edge connectivity
5. Gain knowledge about matrix representation of graph and also find chromatic number of graph.

#### UNIT I

Definition of a Graph – finite & infinite graphs – incidence, degree isolated & pendent Vertices – isomorphisms –sub graphs – walks , paths & circuits –Connected &

disconnected graphs – components –Euler graphs - Operations on Graphs –More on Euler graphs –Hamiltonian paths & circuits .

## UNIT II

Trees –properties of trees –pendent vertices in a tree – distances & centres in a tree – Rooted & binary trees – Spanning trees –Fundamental circuits – Finding all spanning trees of a Graph –Spanning trees in a weighted graph .

## UNIT III

Cut sets – Properties of a Cut set – all Cut sets in a graph – Fundamental circuits & Cut sets –Connectivity & separability.

## UNIT IV

Vector Space of a Graph – Sets with one , two operations –modular arithmetic - Galois Fields –Vectors-Vector Spaces –Basis vectors of a graph – circuit & cutset subspaces – Orthogonal vectors & spaces.

## UNIT V

Matrix representation of a graph – Incidence matrix –Circuit Matrix - Fundamental Circuit Matrix and rank of the circuit matrix – Cut set matrix – adjacency matrix – Chromatic Number - Chromatic partitioning – Chromatic polynomial.

### TEXT BOOK:

Narsingh Deo, Graph Theory with applications to Engineering & Computer Science, Prentice Hall of India ,New Delhi,2<sup>nd</sup> Edition 1997.

### REFERENCES:

- 1.Dr.S. Arumugam & Dr. S. Ramachandran, Invitation to Graph Theory, Scitech Publications India Pvt Limited, Chennai, 2001.
- 2.K.R. Parthasarathy, Basic Graph Theory, Tata McGraw Hill Publishing Company, New Delhi, 1994.
- 3.G.T. John Clark, Derek Allan Holten, A First Look at Graph Theory, World Scientific Publishing company, 1995.

<b>BCS302</b>	<b>JAVA PROGRAMMING</b>				<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
					3	1	0	4

### COURSE OBJECTIVE

- To introduce students to the Java programming language.
- To create Java programs that leverage the object-oriented features of the Java language, such as encapsulation, inheritance and polymorphism; use data types, arrays and other data collections;
- To implement I/O functionality to read from and write to text files.

## **COURSE OUTCOMES**

Upon successful completion of this course, student will be able to

1. Understanding of the principles and practice of object oriented analysis and design in the construction of robust, maintainable programs which satisfy their requirements;
2. Implement, compile, test and run Java programs comprising more than one class, to address a particular software problem.
3. Demonstrate the principles of object oriented programming;
4. Use simple data structures like arrays in a Java program.
5. Understand the concept of package, interface, multithreading and File handling in java.
6. Use members of classes found in the Java API (such as the Math class).
7. Employ various types of selection constructs in a Java program.
8. Employ a hierarchy of Java classes to provide a solution to a given set of requirements.

### **UNIT-I**

Introduction-Object Oriented paradigm-Basic concepts of object oriented programming-Benefits of OOP-Applications of OOP- Java Features- How Java differs from C and C++ – Java and Internet – Java and www – simple Java program –Structure – Java Tokens – Statements

### **UNIT-II**

Constants, Variables, Data Types - Operators and Expressions – Decision Making and Branching: if, if...else, nested if, switch, ?operator - Decision Making and Looping: while, do, for –Jumps in Loops - Labeled Loops – Classes, Objects and Methods.

### **UNIT-III**

Arrays- Introduction-One dimensional arrays –Two dimensional arrays-Strings-Vectors-Enumerated types-Interfaces- Defining, Extending and implementing interfaces- Packages – JAVA API packages- creating packages-Accessing packages-Adding a class to a package-Hiding classes

### **UNIT-IV**

Multithreaded Programming-creating threads- extending thread class- stopping and blocking a thread-life cycle of a thread-synchronization- Implementing a Runnable interface- Inter thread communication-Managing Errors and Exceptions – Applet Programming – Graphics Programming.

### **UNIT-V**

Managing Input/output Files in Java : Concepts of Streams- Stream Classes – Byte Stream classes – Character stream classes – Using streams – I/O Classes – File Class – I/O exceptions – Creation of files – Reading / Writing characters, Byte-Handling Primitive data Types – Random Access Files.

## **TEXT BOOKS:**

1. Programming with JAVA – A PRIMER - E. Balagurusamy, 4<sup>th</sup> Edition, Tata McGraw-Hill, 2010

**REFERENCES:**

1. The Complete Reference JAVA 2 - Patrick Naughton & Hebert Schildt, 3rd ed, TMH
2. Programming with JAVA – John R. Hubbard, 2nd Edition, TMH.
3. JAVA and Object-Oriented Programming Paradigm – Debasish Jana, 2005, PHI.

<b>BCS303</b>	<b>OPERATING SYSTEMS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- The course familiarizes the student with basic knowledge of computer operating systems. The objective of the course is to provide basic knowledge of computer operating system structures and functioning.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Understand the difference between different types of modern operating systems, virtual machines and their structure of implementation and applications.
2. Understand the difference between process & thread, issues of scheduling of user level processes / threads and their issues & use of locks, semaphores, monitors for synchronizing multiprogramming with multithreaded systems and implement them in multithreaded programs.
3. Gain knowledge about the concepts of deadlock in operating systems and how they can be managed / avoided and implement them in multiprogramming system.
4. Demonstrate the design and management concepts along with issues and challenges of main memory, virtual memory and file system.
5. Understand the types of I/O management, disk scheduling, protection and security problems faced by operating systems and how to minimize these problems.

**UNIT – I**

Operating System Objectives and functions-The Evolution of Operating Systems-Serial Processing-Simple batch Systems-Multi Programmed batch Systems-Time Sharing Systems.

**UNIT – II**

Definition of Process-Process States-Process Control Block-Operations on Process-Process Communication-Communication in Client-server System- Basic concepts of threads - Concurrency-Principles of Concurrency-Mutual exclusion - Semaphores – Messages – Deadlock - Deadlock Prevention - Deadlock detection - deadlock avoidance

**UNIT – III**

Memory Management-Address binding-Logical Vs Physical address space-Dynamic Loading-Dynamic Linking and Shared Libraries-Overlays-Swapping-Contiguous Memory

allocation-Paging-Segmentation-Virtual memory-Demand paging-Page replacement-Thrashing.

#### **UNIT – IV**

CPU Scheduling - Scheduling Criteria-Scheduling algorithms – FCFS, SJF, Priority, RR, Multilevel, Feedback Queue - Process synchronization-The Critical Section Problem-Synchronization Hardware-Classical Problems of synchronization, File and Database System-File System-Functions of organization-Allocation and Free space management.

#### **UNIT- V**

Modern Operating Systems-Architecture and Features, Case Studies-Linux –Windows network OS - Windows XP (Design principles and components only)

#### **TEXT BOOK:**

1. Silberschatz, Galvin, Gagne “Operating System Concepts “Sixth Edition-John Willey & Sons INC

#### **REFERENCES:**

1. William Stallings “Operating Systems, Internals and Design Principles”,Fifth Edition-PHI Publications New Delhi
2. Tanenbaum A.S., “Operating System Design & Implementation”, Practice Hall NJ, 2005

<b>BCS304</b>	<b>MICROPROCESSORS AND ITS APPLICATIONS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

#### **COURSE OBJECTIVE**

- To provide a theoretical & practical introduction to microcomputer and microprocessors, assembly language programming techniques, design of hardware interfacing circuit.

#### **COURSE OUTCOME**

Upon successful completion of this subject, student will be able to

1. Solve basic arithmetic operations using the 8085 assembly language.
2. Write program to find out smallest/largest number stored in memory, program related to conversion from Binary to Gray code, Hexadecimal to decimal using the 8085 assembly language.
3. Understand the Hardware and Interfacing.

4. Write program with 8085.
5. Understand BCD Arithmetic, 16-Bit Data operations and Interrupts.
6. Interface with Data Converters
7. Demonstrate the concept of Programmable Interface Devices:

#### **UNIT -I**

Microprocessor – based Systems: Microprocessors, Microcomputers, and Assembly Language, Introduction to Assembly Language Programming. Microprocessor Architecture and Microcomputer Systems.

#### **UNIT -II**

Hardware and Interfacing: 8085 Microprocessor Architecture and Memory Interfacing- Interfacing I/O Devices.

#### **UNIT -III**

Programming the 8085: Introduction to 8085 Instructions –Programming Techniques with Additional Instructions-Counters and Time Delays.

#### **UNIT –IV**

Stack and Subroutines-Code Conversion, BCD Arithmetic, and 16-Bit Data operations- Interrupts.

#### **UNIT- V**

Interfacing Peripherals(I/Os) and Applications: Interfacing Data Converters- Programmable Interface Devices: 8155 I/O Timer, 8279 Keyboard/Display Interface-General-Purpose Programmable Peripheral Devices-Microprocessor Applications.

#### **TEXT BOOKS:**

1. Ramesh S Gaonkar, Microprocessor Architecture, Programming and Application with the 8085, 5th Edition, Penram International Publishing (India) Pvt Ltd, 2008.
2. Mohammed Ali Mazidi and Janice Gillispie Mazidi, The 8051 Microcontroller and Embedded Systems, Pearson Education Asia, New Delhi, 2003.

#### **REFERENCES:**

1. John Uffenbeck, The 80x86 Family, Design, Programming and Interfacing, Third Edition. Pearson Education, 2002.
2. A.K. Ray and K.M.Burchandi, Intel Microprocessors Architecture Programming and Interfacing, McGraw Hill International Edition, 2000.
3. Kenneth J Ayala, The 8051 Microcontroller Architecture Programming and Application, 2nd Edition, Penram International Publishers (India), New Delhi, 1996.
4. M. Rafi Quazzaman, Microprocessors Theory and Applications: Intel and Motorola prentice Hall of India, Pvt. Ltd., New Delhi, 2003.

<b>CHY102</b>	<b>ENVIRONMENTAL SCIENCE</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>2</b>	<b>0</b>	<b>0</b>	<b>2</b>

**COURSE OBJECTIVES**

Creating awareness among engineering students about the importance of environment, the effect of technology on the environment and ecological balance is the prime aim of the course.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Know the importance of environmental studies and methods of conservation of natural resources.
2. Describe the structure and function of an ecosystem.
3. Identify the values and conservation of bio-diversity.
4. Explain the causes, effects and control measures of various types of pollutions.
5. Select the appropriate methods for waste management.
6. Get knowledge about various disaster management methods
7. Recall social issues and legal provision.

**UNIT-I**

Definition, scope, and importance of environmental sciences -Need for public awareness- Natural resources: Forest resources, Water resources, Mineral resources, Energy resources, Land resources - Role of an individual in conservation of natural resources.

**UNIT-II**

Concept of an ecosystem - Structure and function of an ecosystem - Energy flow in the ecosystem - Food chains, food webs and ecological pyramids - Ecological succession - Biodiversity – Definition, value of biodiversity- Hot spots of biodiversity - Threats to biodiversity - Endangered and endemic species of India - Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity

**UNIT-III**

Types, sources, consequences and control measures of water pollution, ecological and biochemical aspects of water pollution - Sources, effects and control measures of Air pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution and Nuclear pollution - Climate change, global warming, acid rain, Tropospheric chemistry of zone-ozone layer depletion, nuclear accidents and holocaust- Role of an individual in prevention of pollution.

**UNIT-IV**

Causes, effects, treatments methods and control measures of solid waste, municipal waste, hazardous waste and biomedical waste - Waste minimization techniques - Cleaner



Technology- Green Chemistry- Principles and its role in controlling environmental pollution- Disaster management: floods, earthquake, cyclone, landslides and Tsunami.

### UNIT-V

Water conservation, rain water harvesting- Resettlement and rehabilitation of people- Wasteland reclamation - Environmental impact assessment- Precautionary and polluters pay principle- environment protection act – air (prevention and control of pollution) act – water (prevention and control of pollution) act – Issues involved in enforcement of environmental legislation - Population explosion – Family Welfare Programmes - Environment and human health - Human Rights - Women and Child Welfare

### TEXT BOOKS

1. Sawyer C. N, McCarty P. L, and Parkin G. F., Chemistry for Environmental Engineering, McGraw-Hill, Inc., New York, 1994.
2. Dhameja, S. K., Environmental engineering and Management, S. K. Kataria and sons, New Delhi, 1<sup>st</sup> edition 2004.
3. Gilbert M.Masters, “Introduction to Environmental Engineering and Science”, pearson education Pvt., Ltd., second edition, ISBN 81-297-0277-0, 2004.
4. Townsend C., Harper J and Michael Begon, “Essentials of Ecology”, Blackwell science.

### REFERENCES

1. Miller T.G. Jr., Environmental science, wadsworth Publishing Co. USA, 2<sup>nd</sup> edition 2004.
2. Bharucha erach, “The Biodiversity of India”, mapin publishing Pvt. Ltd., Ahmedabad India,
3. Trivedi R.K., “Handbook of Environmental Laws”, Rules, Guidelines, Compliances and Standards, Vol. I and II, Enviro media.
4. Cunningham, W.P.Cooper, T.H.Gorhani, “Environmental Encyclopedia”, Jaico Publ., House, Mumbai, 2001.
5. Wager K.D., “Environmental Management”, W.B. Saunders Co., Philadelphia, USA, 1998.
6. Trivedi R.K. and P.K. Goel, “Introduction to air pollution”, techno-science publications.

<b>BCS306</b>	<b>COMPUTER ALGORITHMS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

### COURSE OBJECTIVES

- To study paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.
- It also ensures that students understand how the worst-case time complexity of an algorithm is defined, how asymptotic notation is used to provide a rough classification of algorithms,

- To provide students with a coherent introduction to techniques for using data structures and some basic algorithms, and with the tools for applying these techniques to computational problems.

### **COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Use different computational models (e.g., divide-and-conquer), order notation and various complexity measures (e.g., running time, disk space) to analyze the complexity/performance of different algorithms.
2. Understand the difference between the lower and upper bounds of various problems and their importance in deciding the optimality of an algorithm.
3. Use various techniques for efficient algorithm design (divide-and-conquer, greedy, and dynamic algorithms) and be able to apply them while designing algorithms.
4. Understand and implement various algorithms on graph data structures, including finding the minimum spanning tree and shortest path.
5. Augment various data structures (trees and arrays) to support specific applications.
6. Know various advanced design and analysis techniques such as greedy algorithms, back tracking algorithm, dynamic programming & Know the concepts of Branch & Bound and Number Theoretic Algorithms
7. Understand Graph algorithms, strongly connected components and notion of NP-completeness.

### **UNIT-I**

Growth of functions, Asymptotic notations, Standard notations & common functions, Summations, Recurrences, The efficiency of Algorithms, Analysis of Algorithms.

### **UNIT-II**

Elementary Data Structures – Hash tables, Binary search trees – AVL tree, Red-black tree, B-Trees, Augmenting Data Structures, and Binomial heaps, Fibonacci Heaps, Data Structures for disjoint sets. Divide & Conquer Strategy.

### **UNIT-III**

Dynamic Programming techniques – Principle of Optimality, Matrix – Chain multiplication, Knapsack Problem. Greedy Algorithms – An activity selection problem, Huffman codes, A task scheduling problem.

### **UNIT-IV**

Graph Algorithms – BFS, DFS, Topological sort, strongly connected components, Flow networks, Closest pair of points -Back tracking algorithms.

### **UNIT-V**

Branch & Bound Algorithms. Number Theoretic Algorithms – Matrix multiplication, Inverting matrices, Primality testing, Integer factorization.

### **TEXT BOOK**

1. Thomas H Cormen, Charles E Leiserson, & Ronald L Rivest, 'Introduction to Algorithms', 4<sup>th</sup> Ed. Prentice Hall of India Private Limited, New Delhi, 2008.

## REFERENCES

1. Goodman S E & Hedetniemi, 'Introduction to the Design & analysis of Algorithms', Mc-Graw Hill, 2012.
2. Horowitz E & Sahni S, 'Fundamentals of Computer Algorithms', Galgotia Publications Pvt. Ltd, 2010
3. Aho A V, Ullman, J D & Hopcroft J E, 'The Design and Analysis of Computer Algorithms', Addison Wesley, 2011.
4. Mark Allen Weiss, 'Data Structures and Algorithm Analysis in C++', Pearson, 2011

<b>BCS381</b>	<b>JAVA PROGRAMMING LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	2

## COURSE OBJECTIVES

1. To be knowledgeable enough about basic Java language syntax and semantics to be able to successfully read and write Java computer programs;
2. To have obtained experience designing, implementing, testing, and debugging graphical user interfaces that respond to user events using Java;

## COURSE OUTCOME

Upon successful completion of this course, students will be able to

1. Understand programming language concepts, particularly Java and object-oriented concepts.
2. Write, debug, and document well-structured Java applications
3. Implement Java classes from specifications and effectively create and use objects from predefined class libraries
4. Understand the behavior of primitive data types, object references, and arrays
5. Apply decision and iteration control structures to implement algorithms
6. Write simple recursive algorithms
7. Implement interfaces, inheritance, and polymorphism as programming techniques and apply exceptions handling

## PROGRAMS

1. Programs using classes and methods
2. Programs using one dimensional and two dimensional arrays

3. Programs using Strings
4. Programs using vectors
5. Programs using Inheritance
6. Generate the program using interfaces
7. Programs to implement the exception handling mechanism
8. Programs to implement packages.
9. Programs using multithreading
10. Programs using Applets

<b>BCS382</b>	<b>OPERATING SYSTEMS LABORATORY</b>	L	T	P	C
		0	0	3	2

#### **COURSE OBJECTIVES**

- This Lab course will introduce the basic principles in Operating System. The objective of this course is to enable the students to get practical knowledge in process management, Memory management, File management, Disk management, Network management, I/O management.

#### **COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Write Programs for process creation and synchronization,
2. Write programs for Inter process communication including shared memory, pipes and messages
3. Write Programs using UNIX System calls.
4. Simulate CPU Scheduling Algorithms. (FCFS, RR, SJF, Priority, Multilevel Queuing)
5. Write Program for FIFO, LRU, and OPTIMAL page replacement algorithm

#### **PROGRAMS**

1. Write programs using the following system calls of UNIX operating system: fork, exec, getpid, exit, wait, close, stat, opendir, readdir etc.,
2. Write programs using the I/O system calls of UNIX operating system (open, read, write etc)
3. Write C programs to simulate UNIX commands like ls, grep, etc.
4. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions) Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart

- for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions)
5. Developing Application using Inter Process communication (using shared memory, pipes or message queues)
  6. Implement the Producer – Consumer problem using semaphores (using UNIX system calls).
  7. Implement some memory management schemes – I
  8. Implement some memory management schemes – II
  9. Implement any file allocation technique (Linked, Indexed or Contiguous)

## SEMESTER-IV

<b>BCS401</b>	<b>OPERATIONS RESEARCH</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

### COURSE OBJECTIVES

- The goal of this course is to teach you to formulate, analyze, and solve mathematical models that represent real-world problems.
- To learn linear programming, network flow problems, integer programs, nonlinear programs, dynamic programming and queueing models.

### COURSE OUTCOMES

Upon completion of this course, Student will be able to:

1. Formulate a real-world problem as a mathematical programming model
2. Understand the theoretical workings of the simple method for linear programming and perform iterations of it by hand
3. Understand the relationship between a linear program and its dual, including strong duality and complementary slackness
4. Perform sensitivity analysis to determine the direction and magnitude of change of a model's optimal solution as the data change
5. Solve specialized linear programming problems like the transportation and assignment problems
6. Solve network models like the shortest path, minimum spanning tree, and maximum flow problems
7. Understand the applications of, basic methods for, and challenges in integer programming
8. Understand how to model and solve problems using dynamic programming
9. Model a dynamic system as a queueing model and compute important performance measures

10. Learn optimality conditions for single- and multiple-variable unconstrained and constrained non-linear optimization problems, and corresponding solution methodologies

### **UNIT-I**

Concept -- Meaning and Definition – Development of OR – Characteristics of OR – Scope & Objectives of OR – Phases of OR – Techniques of OR – OR and Modern Business Management- Limitations of OR

### **UNIT-II**

Meaning & Objective of LP – Applications of LP – Formulation of Mathematical Model to a LPP – Objective function – Constraints – Solution to LPP by Graphical method, simplex method and Mi technique – Advantages and Disadvantages of LP.

### **UNIT-III**

Meaning and Objectives – Network Techniques – Managerial applications of Network Techniques – PERT & CPM – Network diagram - – activity – Event – Dummy Activity – Construction of Network diagram – Numbering of events – Activity & Event times – Float & Slack – Steps in the application of CPM – Critical activity – Finding of critical path & Estimated Duration – Time estimates in PERT – Steps involved in PERT calculations – Difference between PERT & CPM (only simple problems are expected – Crashing of activity timing not expected)

### **UNIT-IV**

Solving Assignment Problem- Travelling Salesman Model – Maxima & Minima Method – Hungarian Method Sequential decision making – sequencing problems - Queuing Models.

### **UNIT-V**

Structure of Transportation problem – solution for Transportation problem – North West Corner Method (NWCM) – Least Cost Method (LCM) – Vogel's Approximation method(VOM).

### **TEXT BOOKS:**

1. Kanti Swarup, P.k. Gupta and Manmohan, Operations Research –, Sultan Chand & Sons
2. Paneerselvam R., Operations Research, 4<sup>th</sup> edition, Prentice Hall of India, New Delhi, 2003.

### **REFERENCES**

1. Hamdy A Taha, Introduction to Operations Research, 7<sup>th</sup> edition, Prentice Hall India, New Delhi, 2004.
2. Gupta P.K, Man Mohan, Problem in Operations Research: Methods and Solutions, 9<sup>th</sup> edition, Sultan Chand and Sons, New Delhi, 2003.
3. Vohra N.D., Quantitative Techniques in Management, 2<sup>nd</sup> edition, Tata McGraw Hill, New Delhi, 2001.

4. Kalavathy S., Operations Research, 2nd edition, Vikas Publishing House New Delhi, 2004.

<b>BCS402</b>	<b>WEB TECHNOLOGIES</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

#### **COURSE OBJECTIVE**

- To build web applications using ASP and client side script technologies use with Microsoft's IIS.
- To build XML applications with DTD and style sheets that span multiple domains ranging from finance to vector graphics to genealogy for use with legacy browsers.

#### **COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Understand, analyze and apply the role of languages like HTML, DHTML, CSS, XML, Javascript, VBScript, ASP, PHP and protocols in the workings of the web and web applications
2. Analyze a web project and identify its elements and attributes in comparison to traditional projects.
3. Understand, analyze and create web pages using HTML, DHTML and Cascading Styles sheets.
4. Understand, analyze and build dynamic web pages using JavaScript and VBScript (client side programming).
5. Understand, analyze and build interactive web applications.
6. Understand, analyze and build web applications using PHP.
7. Understand, analyze and create XML documents and XML Schema.

#### **UNIT-I**

World Wide Web – Web browsers – Markup Languages –Style Sheet technologies –client side, server side - HTML – Headings –Links -images- Lists- Tables- Forms- Frames

#### **UNIT-II**

Cascading style sheets-Inline styles-Embedded style sheets-Linking External style sheets- Positioning elements- Dynamic HTML – Object model and collections, Event model, Filters and Transitions.

#### **UNIT-III**

JAVASCRIPT-Introduction – Simple program-Decision making - Equality and Relational operators – Control statements – Functions – Programmer defined functions, JavaScript global functions, Recursion – Arrays – References and Reference parameters, Passing arrays to functions, Multidimensional arrays – Objects – Object types.

**UNIT-IV**

XML-Introduction-Structuring data-XML namespaces-Document Type Definitions (DTDs) and Schema-W3C XML schema documents-XML vocabularies-Document Object Model (DOM), DOM methods- Simple API for XML (SAX)-Extensible Style sheet Language (XSL)-Simple Object Access Protocol (SOAP).

**UNIT-V**

PHP-Introduction-String processing and regular expressions-Viewing Client/Server environment variables-Form processing and Business logic-Verifying a username and password-connecting to a database

**TEXT BOOKS**

1. Deitel, Deitel and Neito, INTERNET and WORLD WIDE WEB – How to program, Pearson Education Asia, 5<sup>th</sup> Edition , 2011.

**REFERENCES**

1. Achyut S Godbole and Atul Kahate, “Web Technologies”, Second Edition, Tata McGraw Hill, 2012.
2. Thomas A Powell, Fritz Schneider, “JavaScript: The Complete Reference”, Third Edition, TataMcGraw Hill, 2013.
3. David Flanagan, “JavaScript: The Definitive Guide, Sixth Edition”, O'Reilly Media, 2011

<b>BCS403</b>	<b>DATABASE MANAGEMENT SYSTEMS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

**COURSE OBJECTIVES**

- To educate students with fundamental concepts of Data Base Management System, Data Models, Different Data Base Languages.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Analyze Database design methodology.
2. Acquire knowledge in fundamentals of Data Base Management System.
3. Analyze the difference between traditional file system and DBMS.
4. Handle with different Data Base languages.
5. Draw various data models for Data Base and Write queries mathematically.
6. Design data base and normalize data and Understand how query are being processed and executed.
7. Deal with online transactions and control Concurrency.
8. Understand types of Data Base failures and Recovery.

**UNIT – I**



Introduction – Database- Database management system- Characteristics of the database approach- Role of Database administrators- Role of Database Designers- End Users- Categories of data models- Schemas- Instances- - DBMS Architecture and Data Independence – The Three schema architecture- DBMS Languages and Interfaces- Classifications of Database Management Systems.

## UNIT II

Data Modelling Using Entity-Relationship Model -Using High Level Conceptual Data Models for Database Design- Example Database applications. Entity types- Entity Sets- Attributes and Keys. Relationships- Relationship types- Roles and Structural constraints. Weak Entity Types and Drawing E- R Diagrams

## UNIT III

Database Design -Functional dependencies and Normalization for Relational Databases - Normalization concepts- first- second- third normal forms

## UNIT IV

SQL data definition and data types- specifying constraints in SQL- schema change statements- Basic queries- INSERT- DELETE and UPDATE statements in SQL- Views – Concept of a view in SQL.

## UNIT - V

Transaction Processing Concepts and Concurrency Control Techniques -Transaction and System concepts – Desirable properties of Transactions – Schedules and Recoverability. Lock-Based Protocols – Locks- Granting of Locks- and Two phase locking protocol.

## TEXT BOOK

1.Elmasri & Navathe, Fundamentals of Database Systems,5<sup>th</sup> Edition, Addison Wesley 2008.

## REFERENCES

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6<sup>th</sup> Edition, McGraw Hill, 2011.
- 2.Patrick O'Neil, Elizabeth O'Neil, Database Principles Programming and Performance, 2<sup>nd</sup> Edition, Margon Kaufmann Publishers India ,2001

<b>BCS404</b>	<b>COMPUTER NETWORKS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

## COURSE OBJECTIVES

- To be familiar with the basics of data communication;

- To be familiar with various types of computer networks;
- To have experience in designing communication protocols;
- To be exposed to the TCP/IP protocol suite.

### **COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Define, use and implement Computer Networks and the basic components of a Network system.
2. Know and Apply pieces of hardware and software to make networks more efficient, faster, more secure, easier to use, able to transmit several simultaneous messages, and able to interconnect with other networks.
3. Differentiate the various types of network configurations and applying them to meet the changing and challenging networking needs of organizations.
4. Understand the layers of OSI and TCP and get knowledge about congestion control and network security
5. Define the different protocols, software, and network architectures.
6. Define the concept of local area networks, their topologies, protocols and applications.
7. Analyze why networks need security and control, what errors might occur, and how to control network errors.

### **UNIT-I**

Introduction-Uses of Computer Networks-Network Hardware-Network Software-Reference Models –Example Networks-Network Standardization-Metric Units.

### **UNIT-II**

The Physical Layer: The Theoretical basis for Data Communication – Guided Transmission Media- Wireless Transmission-The Public Switched Telephone Network-Cable Television.

### **UNIT-III**

The Data Link Layer: Data Link Layer Design Issues-Error Detection and Correction-Elementary Data Link Protocols-Sliding Window Protocols-Protocol Verification.-Example Data Link Protocols. MAC: The Channel Allocation Problem-Multiple Access Protocols-Ethernet-Wireless LANS.

### **UNIT-IV**

The Network Layer: The Network Layer Design Issues-Routing Algorithms-Congestion Control Algorithms-QoS-Internetworking. The Transport Layer: Elements of Transport Protocols-UDP-TCP.

### **UNIT -V**

The Application Layer DNS- Electronic Mail- WWW- Multimedia. Network Security: Cryptography- Symmetric Key Algorithms, Public Key Algorithms, Digital Signatures Authentication Protocols –Communication Security- Web Security.

### TEXT BOOKS

1. Andrew S Tanenbaum , Computer Networks, 6th Ed., Pearson Education 2009.

### REFERENCES

1. B.MuthuKumaran, Computer Networking, 3<sup>rd</sup> Ed., TMG 2009.
2. Behrouz A Forouzan, ‘Data Communications and Networking’, 4th Ed., TMH 2008.
3. Uyles Black, ‘Data Networks: concepts theory and practices’, 4<sup>th</sup> Ed PHI, 1998.

<b>BCS405</b>	<b>PC HARDWARE AND TROUBLE SHOOTING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

### COURSE OBJECTIVES

- The course is made up of the following major area: Operating Systems, Security, Mobile Devices, and Troubleshooting.

### COURSE OUTCOME

Upon successful completion of this course, students will be able to

1. Understand basic concept & structure of computer hardware & networking
2. Identify the existing configuration of the computers and peripherals and Upgrading same as and when required
3. Apply their knowledge about computer peripherals to identify / rectify problems onboard
4. Integrate the PCs into Local Area Network & re-install operating systems and various shipboard applications.
5. Perform routine maintenance, upgrades
6. Manage data backup & restore operations on server and update anti-virus software and set schedules

### UNIT - I

CPU essentials – processor modes – modern CPU concepts – Architectural performance features – the Intel’s CPU – CPU over clocking – over clocking requirements – over clocking the system – over clocking the Intel processors – Essential memory concepts – memory organizations – memory packages – modules – logical memory organizations – memory considerations –memory types – memory techniques – selecting and installing memory.

## **UNIT -II**

Active motherboards – sockets and slots – Intel D850GB – Pentium4 mother board – expansion slots – form factor – upgrading a mother board – chipsets – north bridge – south bridge – CMOS – CMOS optimization tactics – configuring the standard CMOS setup – motherboard BIOS – POST – BIOS features – BIOS and Boot sequences – BIOS shortcomings and compatibility issues – power supplies and powermanagement – concepts of switching regulation – potential power problems – power management.

## **UNIT-III**

The floppy drive – magnetic storage – magnetic recording principles – data and disk organization – floppy drive – hard drive – data organization and hard drive – sector layout – IDE drive standard and features – Hard drive electronics – CD-ROM drive – construction – CDROM electronics – DVD-ROM – DVD media – DVD drive and decoder.

## **UNIT-IV**

Parallel port – signals and timing diagram – IEEE1284 modes – asynchronous communication - serial portsignals – video adapters – graphic accelerators – 3D graphics accelerator issues – DirectX – mice – modems – keyboards – sound boards .

## **UNIT-V**

Buses – Industry standard architecture (ISA), peripheral component Interconnect (PCI) – AcceleratedGraphics port (AGP) – plug-and-play devices – SCSI concepts – USB architecture.

## **TEXTBOOK:**

1. Stephen J.Bigelow, “Trouble Shooting, maintaining and Repairing PCs”, Tata McGraw-Hill, New Delhi, 2001.

## **REFERENCES:**

1. Craig Zacker & John Rourke, “The complete reference:PC hardware”, Tata McGraw-Hill, New Delhi, 2001.
2. Mike Meyers, “Introduction to PC Hardware and Trouble shooting”, Tata McGraw-Hill, New Delhi, 2003.
3. B.Govindarajulu, “IBM PC and Clones hardware trouble shooting and maintenance”, Tata McGraw-Hill, New Delhi, 2002.

<b>BCS481</b>	<b>WEB TECHNOLOGIES LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	2

**COURSE OBJECTIVES**

- On completion of this course, a student will be familiar with client server architecture and able to develop a web application using java technologies To create fully functional website/web applications.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Understand, analyze and apply the role of languages like HTML, DHTML, CSS, XML, Javascript, VBScript, ASP, PHP and protocols in the workings of the web and web applications
2. Analyze a web page and identify its elements and attributes.
3. Create web pages using HTML, DHTML and Cascading Styles sheets.
4. Create dynamic web pages using JavaScript and VBScript (client side programming).
5. Create interactive web applications using ASP.NET.
6. Build web applications using PHP.
7. Create XML documents and XML Schema.
8. Build and consume web services.

**PROGRAMS****1.HTML and JavaScript**

- a) Preparation of Bio-data using Forms in HTML.
- b) Simple Calculation
  - a. Inventory Calculation.
- c) Input Validation : Payroll maintenance
- d) Event Handling
  - a. Changing the Background Color of the Window.
- e) Develop a Dynamic Web page Using CSS properties and elements for a university website
- f) To generate the random numbers and display in a table format.
- g) Generation of Fibonacci series
- h) Different Pascal triangle generation
- i) Function to determine the pair of integers whether the second integer is multiple of the first.
- j) Quiz program
- k) Create a guessing number game
- l) HTML fom validation

**2.PHP and MYSQL**

- a) Program to implement the concept of operator, arrays and functions
- b) Program to communicate between two web pages using PHP.
- c) Program to create session and cookies
- d) Program for file manipulation in PHP
- e) Create a database with two tables in MYSQL and perform the basic query operations.
- f) Demonstration of joining tables and usage of sub queries.
- g) Working with string, numeric and date functions in MYSQL.
- h) Develop a application for the demonstration of database connectivity to PHP with MySQL.
- i) Develop a simple application for student academic performance

<b>BCS482</b>	<b>RDBMS LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	3

**COURSE OBJECTIVES**

- To educate students with fundamental concepts of Data Base Design, Data Models, Different Data Base Languages (SQL/Oracle).
- To analyze Data Base design methodology and DB connectivity.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Gain knowledge about SQL Fundamentals.
2. Perform Unary & Binary table operations.
3. Handle with different Data Base languages.
4. Create Table View, Log & Triggers.
5. Handle online Transactions.
6. Create Database connectivity with front-end.
7. Write Embedded and Nested Queries.
8. Create index and views
9. Create procedures, Triggers and cursors

**PROGRAMS**

1. Queries Using DDL- DML commands
2. Queries using AND- OR- NOT operation , Union- Intersection and Projection
3. Join Operation
4. Sorting and Grouping
5. Nested queries using SQL
6. Built-in functions of SQL
7. Update operations using SQL
8. Use of SQL forms
9. Use of indexes- creating view and querying in views

10. Use procedures for Income Tax calculation
11. Use Cursor for Pay -roll system

<b>BCS483</b>	<b>COMMUNICATIVE ENGLISH LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	2	1

### **COURSE OBJECTIVES**

- Ensure of understanding of the basics of communication through English, application of the various models of verbal and non-verbal communication in the social and professional sphere, the rules of phonology and its application, understanding the basics of grammar to improve communication and speak a neutral and correct form of English.

### **COURSE OUTCOMES**

Upon Successful completion of this course, students will be able to

1. Acquire knowledge about the various principles of communication, understand its various stages and the role of audience and purpose, deal with the barriers that affect communication in a professional set up.
2. Understand the different channels that are functional at the work place.
3. Learn the importance of verbal and non-verbal communication in the professional world along with its uses.
4. Learning the uses and application of RP to improve pronunciation.
5. Understanding the importance of intonation, word and sentence stress for improving communicative competence, identifying and overcoming problem sounds.
6. Importance of syntax for cultivating effective language skills..
7. Review of grammar – verbs and its different forms and application of the different forms of advanced grammar.
8. Imbibing the knowledge of effective classroom speaking and presentation.
9. Learning the nuances of effective writing by using short and crisp sentences.
10. Understanding and adhering to deadlines while completing assignments.

Module-1: (Phonetics Theory & Practice)

Module-2: (Communicative English-Interactive)

Module-3: (Situational Conversation-Listen)

Module-4: (Situational Conversation-Listen)

Module-5: (Situational Conversation-Listen)

Module-6: (Global Communication-Listen)

Module-7: (Global Communication-Listen)

Module-8: (Monologues with exercise)

Module-9: (Comprehension)

Module-10: (Biographies)

- Module-11: (Errors in Spoken English)
- Module-12: (Essential English Grammar)
- Module-13: (Vocabulary)
- Module-14: (Learning Through Cartoons)
- Module-15: (Learning Through Fun)
- Module-16: (Learning Through Games)
- Module-17: (Learning Through Music)
- Module-18: (Learning Through Stories)
- Module-19: (Learning Through Situations)
- Module-20: (Learning Through Activities)
- Module-21: (Learning Through Discussion)
- Module-22: (Learning Through Interview)

## **SEMESTER-V**



<b>BCS501</b>	<b>SYSTEM SOFTWARE</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- To view some of the major tasks of the system software of a computer system, focusing on internal working of the hardware and software interface of a typical system.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Identify and understand different components of a compiler and their functioning.
2. Know lexical, syntax and semantic analysis processes.
3. Understand and define the role of lexical analyzer, use of regular expression and transition diagrams.
4. Understand Finite state machine and purpose
5. Know, and use Context free grammar, and parse tree construction
6. Understand the working of lex and yacc compiler for debugging of programs
7. Comprehend and apply error detection and correction methods
8. Determine code generation and optimization techniques

**UNIT - I**

System software and machine architecture – The Simplified Instructional Computer (SIC) - Machine architecture - Data and instruction formats - addressing modes - instruction sets - I/O and programming.

**UNIT -II**

Basic assembler functions - A simple SIC assembler – Assembler algorithm and data structures - Machine dependent assembler features - Instruction formats and addressing modes – Program relocation - Machine independent assembler features - Literals – Symbol-defining statements – Expressions - One pass assemblers and Multi pass assemblers - Implementation example - MASM assembler.

**UNIT-III**

Basic loader functions - Design of an Absolute Loader – A Simple Bootstrap Loader - Machine dependent loader features - Relocation – Program Linking – Algorithm and Data Structures for Linking Loader - Machine-independent loader features - Automatic Library Search – Loader Options - Loader design options - Linkage Editors – Dynamic Linking – Bootstrap Loaders - Implementation example - MSDOS linker.

**UNIT -IV**

Basic macro processor functions - Macro Definition and Expansion – Macro Processor Algorithm and data structures - Machine-independent macro processor features - Concatenation of Macro Parameters – Generation of Unique Labels – Conditional Macro

Expansion – Keyword Macro Parameters-Macro within Macro-Implementation example - MASM Macro Processor – ANSI C Macro language.

### UNIT-V

Text editors - Overview of the Editing Process - User Interface – Editor Structure. - Interactive debugging systems - Debugging functions and capabilities – Relationship with other parts of the system – User-Interface Criteria.

### TEXT BOOK :

1. Leland L. Beck, “System Software – An Introduction to Systems Programming”, 4<sup>th</sup> Edition, Pearson Education Asia, 2010.

### REFERENCES:

1. D. M. Dhamdhere, “Systems Programming and Operating Systems”, Second Revised Edition, Tata McGraw-Hill, 2009.
2. John J. Donovan “Systems Programming”, Tata McGraw-Hill Edition, 2009.
3. John R. Levine, Linkers & Loaders – Harcourt India Pvt. Ltd., Morgan Kaufmann Publishers,2009

<b>BCS502</b>	<b>SOFTWARE ENGINEERING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

### COURSE OBJECTIVES

- The aim of the course is to assist the student in understanding the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.

### COURSE OUTCOMES

Upon successful completion of this course, students will be able to

1. Select and implement different software development process models
2. Extract and analyze software requirements specifications for different projects
3. Develop some basic level of software architecture/design
4. Apply standard coding practices
5. Define the basic concepts and importance of Software project management concepts like cost estimation, scheduling and reviewing the progress.
6. Identify and implement of the software metrics
7. Apply different testing and debugging techniques and analyzing their effectiveness.
8. Analyze software risks and risk management strategies
9. Define the concepts of software quality and reliability on the basis of International quality standards.

### **UNIT - I**

Software-Characteristics, Classification, Myths, Crisis, Software Engineering: Definition, Comparison with other disciplines, Ethics & professional practice , Phases in Software Engineering, Challenges, Software Process, Project, Product - Components of Software process ,process framework, process assessment , Software Life Cycle Models , Selection criteria, Process change management , Quantitative process management

### **UNIT-II**

Software Requirements – Definition, Types, Requirement Engineering process, Feasibility Study - Types of feasibilities, Process Requirements Elicitation - techniques, Requirements Analysis – Structured Analysis, Object Oriented Modeling, Other approaches, Requirements Specification – Structure of SRS, Requirements Validation , Requirements Management – A Case study

### **UNIT-III**

Software Design – basic principles, concepts , Data design , Data Architectural design, Component level design , User Interface design ,Pattern based Software design, Design Notations, Design Reviews – types, process, evaluating reviews, Software Design Documentation, A Case study, Software Coding – features, guidelines, Methodology, Programming practices, Verification techniques, documentation

### **UNIT-IV**

Software Testing - basics, guidelines, characteristics, Test Plan – steps in development, Software testing strategies, V Model of Software testing, Levels of Software testing – Unit, Integration, System, Acceptance, Testing Techniques (basic idea of black box and white box testing), Object Oriented testing, Debugging, Software test report, Software Maintenance – basics, Legacy Systems, factors affecting maintenance, types of maintenance, Life cycle, Models, Techniques

### **UNIT – V**

Software Planning and Scheduling – project planning, planning process, project plan, Project Scheduling – principles, techniques, Project staffing , Risk management, Software Quality – Concepts, Quality Assurance Activities, Software reviews, Evaluation, Capability Maturity Model , Software Reliability, Software Configuration Management process, Concept of Software Re Engineering – approaches, process models.

### **TEXT BOOK :**

1. Software Engineering Principles And Practices By Rohit, Khurana, Vikas Publishing House Pvt. Ltd. 2<sup>nd</sup> Edition,2010

### **REFERENCES :**

1. Software Engineering A Practitioners Approach By ROGER S, Pressman 6<sup>th</sup> Edition Mcgraw Hill International Edition,2005
2. Rajib Mall, Fundamentals of Software Engineering, PHI,3<sup>rd</sup> Edition,2009

<b>BCS503</b>	<b>OPEN SOURCE TECHNOLOGIES</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	1	0	4

**COURSE OBJECTIVES**

- The objective of this course is to utilize and contribute to open source projects. To make the students to gain experience using open source tools, languages and frameworks to prepare for careers in software development.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Understand the Open source Principles and Free software
2. Get knowledge about the methodology and Languages used to develop open source products
3. Demonstrate the Infrastructure services
4. Ability to understand the concept of Wordpress – Moodle – Android Application Development
5. Acquire knowledge about open source desktop and different type of vendors

**UNIT-I**

Introduction- Open source Principles – Open Standard requirements for software- Open source Successes – Free Software – Free Software Vs. Open Source Software – public domain software- Proprietary Vs. Open source Licensing Model- Open Standards Principles – methodologies – open source software development model – Legal Issues in open source

**UNIT-II**

Methodology – Languages used to develop open source products – cross platform code – implementation roles – open source impact on team issues – implementation process – implementation principles – key documents – migration – interacting with the open source community - support

**UNIT-III**

Contents of the Operating system – Linux distribution vendors – Enterprise Distribution vendors – Community Supported Distribution vendors – international alternatives – Introduction to open source desktop – graphical desktops – web browsers – the office suite- mail and calendar clients – personal software

**UNIT-IV**

Infrastructure services – web servers – database servers – mail servers – Perl – PHP – Python – Ruby – Wordpress – Joomla – Drupal – Moodle - Android

**UNIT-V**

Linux Operating System – Wordpress – Moodle – Android Application Development .

**TEXT BOOKS:**

1. Kailash vedera, Bhavyesh Gandhi “Open Source Technology” , University Science Press, New Delhi

**REFERENCE**

1. Paul Kavanagh, “Open Source Software: Implementation and Management”, Elsevier Digital Press
2. Kailash Vadera, Bhavyesh Gandhi “Open Source Technology”, Laxmi Publications

<b>BCS504</b>	<b>COMPUTER GRAPHICS</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	1

**COURSE OBJECTIVES**

- This course is designed to provide a comprehensive introduction to computer graphics leading to the ability to understand contemporary terminology, progress, issues, and trends.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Demonstrate an understanding of contemporary graphics hardware.
2. Create interactive graphics applications in C++ using one or more graphics application programming interfaces.
3. Write program functions to implement graphics primitives.
4. Write programs that demonstrate geometrical transformations.
5. Demonstrate an understanding of the use of object hierarchy in graphics applications.
6. Write program functions to implement visibility detection.
7. Write programs that demonstrate computer graphics animation.
8. Write programs that demonstrate 2D image processing techniques.

**UNIT – I**

Overview of Graphic Systems – Display Devices – hard copy Devices – Interactive Input Devices – Display Processor – Graphic software – Output Primitives – Line Drawing Algorithms – Initialising Lines – Line command – fill areas – circle Generation Algorithms.

**UNIT – II**

Attributes of output primitives – line style – color and Intensity – area filling algorithms – character Attributes – inquiry functions – bundled attributes – two dimensional transformations – basic and composite transformations – metric representations.

**UNIT – III**

Windowing and Clipping – Windowing concepts – Clipping Algorithms – Window to view port Transformations – segments – Interactive input methods – Physical input devices –

logical classification of input devices – interactive picture construction techniques – input functions.

#### **UNIT – IV**

Three dimensional concepts – 3D Display Techniques – 3D representation – polygon and curved surface – 3D transformations.

#### **UNIT – V**

3D viewing – projections – viewing transformation –Implementation of viewing operations – Hidden surface and Hidden Line removal – back free removal, depth buffer and scan line methods – shading.

#### **TEXT BOOK:**

1. Donald Hearn and M.Pualine Baker “Computer Graphics”,PHI , 2<sup>nd</sup> Edition.

#### **REFERENCE**

1. William M.Neuman and Robert F Sproul “Principles of Interactive computer Graphics” , McGraw Hill International Edition,2<sup>nd</sup> Edition.
2. Foley, van Dam, Feiner, and Hughes. Computer Graphics: Principles and Practice, 3rd edition in C.

<b>BCS581</b>	<b>OPEN SOURCE TECHNOLOGIES LABORATORY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		0	0	3	1

#### **COURSE OBJECTIVES**

- Open source laboratory introduces the students to develop technical solutions for problems using the open source software’s readily available at free of cost. The objective of this course is to enhance the knowledge of students to address the IT requirements both from the operating system and application requirement perspective.

#### **COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Work on open source OS ubuntu.
2. Select and install the various open source software as per the requirement.
3. Construct the development environment as per the requirement.
4. Identify the associated plug-in’s as per the requirement.
5. Learn the languages like PHP, Gambas, Perl, Python to develop an application as per requirement.
6. Create the virtualization environment.
7. Work on user space drivers using FUSE.
8. Manage the versions of the software using VCS tools.

**PROGRAMS**

- 1.Developing Applications using PHP
- 2.Developng Applications using JOOMLA
- 3.Developing Applications using Andriod
- 4.Developing Applications using PHYTHON
- 5.Developing Appliaitions using WordPress.

<b>BCS582</b>	<b>GRAPHICS LABORATORY</b>	L	T	P	C
		0	0	3	1

**COURSE OBJECTIVES**

1. Main objective of lab to measure the students' understanding of the Computer Graphics techniques concepts and algorithm.
2. To implement of line drawing, circle drawing, polygon drawing, transformation of objects, scaling, viewing, and curve designing and modeling algorithm practically for graphics.
3. To motivate students need to think critically and creatively in order to come with an alternate solution for an existing problem with computer graphics.

**COURSE OUTCOME**

Upon successful completion of this course, students will be able to

1. Understand practical fundamental of line drawing, circle drawing, polygon drawing and curve drawing.
2. Understand the concepts of different type of geometric transformation of objects in 2D and 3D.
3. Understand the practical implementation of modeling, rendering, viewing of objects in 2D and 3D.
4. Get knowledge about clipping algorithms
5. List out the shapes and filling algorithms
6. Create Presentations using Multimedia tool
7. Perform Animation and Morphing using Multimedia tool

**PROGRAMS**

1. Generating lines using line drawing algorithms.
2. Generating circles using circle generation algorithms .
3. Generating ellipse using ellipse generation algorithms.
4. Filling shapes using filling algorithms.
5. Scaling, rotating and translating an object in 2D.
6. Clipping a line using line clipping algorithm.

7. Clipping a polygon using polygon clipping algorithm.
8. Animation program.
9. Creating a presentation using a multimedia tool.
10. Animation and Morphing using multimedia tool.



**SEMESTER VI**

<b>BCS601</b>	<b>VISUAL PROGRAMMING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- This course introduces computer programming using the Visual BASIC programming language with object-oriented programming principles. Emphasis is on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test and debug at a beginning level.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Design, create, build, and debug Visual Basic applications.
2. Explore Visual Basic's Integrated Development Environment (IDE).
3. Implement syntax rules in Visual Basic programs.
4. Explain variables and data types used in program development.
5. Apply arithmetic operations for displaying numeric output.
6. Write and apply decision structures for determining different operations.
7. Write and apply loop structures to perform repetitive tasks.
8. Write and apply procedures, sub-procedures, and functions to create manageable code.
9. Create one and two dimensional arrays for sorting, calculating, and displaying of data.
10. Write Visual Basic programs using object-oriented programming techniques including classes, objects, methods, instance variables, composition, and inheritance, and polymorphism.
11. Write Windows applications using forms, controls, and events.

**Unit I**

Visual Basic – Variables – Constant – Arrays – Collections – Procedures – Subroutines, Functions, Calling Procedures – Arguments – Argument Passing Mechanisms, Using Optional Arguments, Passing an UnKnown Number of Arguments, Named Arguments – Function Return Values – Returning Custom Data Type, Arrays, Error as Function Values – Control Flow Statements – If..Then, If..Then..Else, Select Case – Loop Statements – Do..Loop, For..Next, While..Wend – Nested Control Structures – The Exit Statement.

## **Unit II**

Working With Forms: The Appearance of Forms – Start up, Loading, Showing and Hiding, Controlling – Designing Menus – Menu Editor, Programming Menu Commands, Using Access and Shortcut Keys, Manipulating Menus at Runtime – Building Dynamic form at Runtime.

Basic ActiveX Controls – The TextBox Control – Basic Properties, Manipulating Control's Text, Text Selection, Search and Replace Operations, Capturing KeyStrokes – The ListBox and ComboBox Control – Basic Properties, Control's methods, Indexing with the ListBox – Searching Sorted List, ComboBox Control – The ScrollBar and Slider Controls – ScrollBar Control, Slider Control – File Controls.

## **Unit III**

Drawing with Visual Basic: Graphics Controls – Sizing Images, Loading and Saving Images, Setting Picture and Image Properties, Exchanging through Clipboard – Coordinate Systems – Scale properties and Methods, Twips Per Pixel X, Twips Per Pixel Y Properties, Current X Current Y Properties – Graphics Methods – Drawing Text, Line and Shapes, Filling Shapes, Circle Method, Drawing Modes, Drawing Curves – Optimization Issues.

## **Unit IV**

Advanced ActiveX Controls: The Common Dialogs Control – Usage, Properties, Color, Font, FileOpen and FileSave Common Dialog Box, Multiple File Selection, Print and Help Common DialogBox – TreeView and List View – How Tree Structure work, ImageList, Using TreeView and List Control, Structuring Tree View Control, Viewing Folder's Files.

More Advanced ActiveX Controls: RichTextBox Control – RTFLanguage, Text Manipulation Properties, RichTextBox Control's Methods TextFormatting Properties – MSFlexGrid Control– Basic Properties, Data Entry – Multiple Document Interface – Basics, Built-in Capabilities, Parent and Child Menus – Accessing Child Forms.

## **Unit V**

DataBase Programming: RecordSets, Data Control, Data Control's Properties, Data Control's Methods – Understanding Relational Concepts – Using Visual Data Manager – Structure of the BIBLIO DataBase – Validating Data – Entering Data – Accessing fields in Recordset – Introduction to SQL – Advanced Data – Bound Controls.

Active Data Objects: Creating Data Project – Designing with DataEnvironment ActiveX Designer – Designing Command Hierarchies, DataEnvironment with DataGrid Control and MSFlexGrid Control, DataReport ActiveX Designer – ADO Data Control – Programming the Active Data Objects, ADO Object Model, Using ADO, Establishing a Connection, Executing SQL Statements, Manipulating the Recordset Object, Record Editing and Updating.

**Text Book**

Evangelos Petroustos, Mastering Visual Basic 6, BPB Publications, New Delhi.

**Reference**

1. PK.MCBride, Programming in Visual Basic, BPB Publications, New Delhi.
2. Steve Brown, Visual Basic 6 in Record Time, BPB Publications.
3. Gary Cornell & Troy Strain, Visual Basic Nuts & Bolts For Experienced Programmers, MCGrawHill Publication, New Delhi.

<b>BCS602</b>	<b>SOFTWARE TESTING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES**

- The objective of this course is to make the student to learn the purpose of testing, Path testing, Data flow testing, domain testing, regular expressions and flow anomaly, Logic based testing.

**COURSE OUTCOMES**

Upon completion of the subject, students will be able to

1. Understands the process to be followed in the software development life cycle
2. Find practical solutions to the problems
3. Solve specific problems alone or in teams
4. Manage a project from beginning to end
5. Work independently as well as in teams
6. Define, formulate and analyse a problem

**UNIT-I**

Human and errors- Testing and Debugging- Software Quality-Requirement-Behavior and Correctness- Fundamentals of Test Process- Psychology of Testing- General Principles of Testing- Test Metrics

**UNIT-II**

Review of software development models (Waterfall Models, Spiral Model, W Model-V Model) - Agile Methodology and Its Impact on testing- Test Levels (Unit, Component, Module, Integration, System, Acceptance, Generic)

**UNIT-III**

Static Testing- Structured Group Examinations -Static Analysis -Control flow & Data flow-Determining Metrics -Dynamic Testing -Black Box Testing -Equivalence Class Partitioning, Boundary Value Analysis, State Transition Test- Cause Effect Graphing and Decision Table Technique and Used Case Testing and Advanced black box techniques -

White Box Testing -Statement Coverage, Branch Coverage, Test of Conditions, Path Coverage, Advanced White Box Techniques, Instrumentation and Tool Support -Gray Box Testing- Intuitive and Experience Based Testing

#### UNIT-IV

Test Organization -Test teams, tasks and Qualifications -Test Planning Quality Assurance Plan- Test Plan, Prioritization Plan- Test Exit Criteria Cost and economy Aspects -Test Strategies - Preventive versus Reactive Approach, Analytical versus heuristic Approach -Test Activity-Management, Incident Management, Configuration Management -Test Progress Monitoring and Control -Specialized Testing: Performance, Load, Stress & Security Testing

#### UNIT-V

Automation of Test Execution, Requirement tracker, High Level Review Types of test Tools -Tools for test management and Control, Test Specification, Static Testing -Dynamic Testing, Non functional testing

#### TEXT BOOK:

1.Glenford J.Myers,"The Art of Software Testing",Second edition,John Wiley sons

#### REFERENCE:

1. Aditya P. Mathur, "Foundations of Software Testing", Addison-Wesley Professional,2008
2. Ron Patton "Software Testing" Second Edition, Pearson Education
3. Naresh Chauhan " Software Testing Principles and Practices, Oxford University Press

<b>BCS681</b>	<b>VISUAL PROGRAMMING LABORATORY</b>	L	T	P	C
		0	0	3	1

#### COURSE OBJECTIVES

- This course introduces computer programming using the Visual BASIC programming language with object-oriented programming principles. The objective of this course is to make the student to learn how to design, code, test and debug programs using VC++ and VB.

#### COURSE OUTCOMES

Upon successful completion of this lab course, students will be able to

1. Design, create, build, and debug Visual Basic applications.
2. Apply arithmetic operations for displaying numeric output.
3. Apply decision structures for determining different operations.
4. Write and apply loop structures to perform repetitive tasks.
5. Write and apply procedures, sub-procedures, and functions to create manageable code.
6. Create one and two dimensional arrays for sorting, calculating, and displaying of data.

7. Write Visual Basic programs using object-oriented programming techniques including classes, objects, methods, instance variables, composition, and inheritance, and polymorphism.
8. Write Windows applications using forms, controls, and events.

### PROGRAMS

1. Write a visual basic program to find the sum of digits till a single digit
2. Write a visual basic program to find the mid character of a string
3. Write a visual basic program to find the simple interest and compound interest
4. Write a visual basic program to check whether a given string is a palindrome or not
5. Write a visual basic program to find the binomial coefficient for a given N and R
6. Write a visual basic program to arrange the numbers in order
7. Write a visual basic program to the names in alphabetic order
8. Write a visual basic program to perform arithmetic operation using stack
9. Write a visual basic program to add and delete an item from a list
10. Write a visual basic program to transfer an item from one list to another list
11. Write a visual basic program to implement all time functions
12. Write a visual basic program to design a calculator using control arrays
13. Write a visual basic program to find day of the date after 'n' days from current date
14. Write a visual basic program to draw different shapes and fill them
15. Write a visual basic program to create your own note pad
16. Write a visual basic program to implement graphic functions and also to include free hand drawing.
17. Write a visual basic program to animate a picture
18. Write a visual basic program to prepare a mark statement for the students
19. Write a visual basic program to prepare the Electricity bill
20. Write a visual basic program to prepare the pay bill.

<b>BCS682</b>	<b>SOFTWARE TESTING LABORATORY</b>			
	L	T	P	C
	0	0	3	1

### COURSE OBJECTIVES

- To demonstrate the UML diagrams for various applications
- To demonstrate the working of software testing tools.
- To Study of testing tools- win runner, selenium etc.
- To Write test cases for various applications

### COURSE OUTCOMES

Upon successful completion of this course, students will be able to

1. understands the process to be followed in the software development life cycle
2. find practical solutions to the problems
3. solve specific problems alone or in teams

4. manage a project from beginning to end
5. work independently as well as in teams
6. define, formulate and analyze a problem

#### **PROGRAMS**

1. Study of various tools for software testing such as WinRunner, LoadRunner, Rational Rose Test Suite etc.,
2. Performing the following testing using the testing tools
  - a. Requirements testing
  - b. use-case scenario testing
  - c. Design testing
  - d. Code testing
  - e. Path testing
  - f. Code coverage testing
  - g. Data-flow testing
  - h. Load testing
  - i. Regression testing
  - j. Documentation testing
3. Developing an automated test-case generation tool for domains such as :
  - a. web-site development
  - b. inventory management
  - c. shopping cart

**FIFTH SEMESTER ELECTIVES**

<b>BCS511</b>	<b>ADVANCED DATABASES</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES**

- This course aims to give students in depth information about system implementation techniques, data storage, representing data elements, database system architecture, the system catalog, query processing and optimization, transaction processing concepts, concurrency control techniques, database recovery techniques, database security and authorization, enhanced data models for advanced applications, temporal databases, deductive databases, database technology for decision support systems, distributed databases and client server architecture, advanced database concepts, and emerging technologies and applications.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Understand object oriented and persistent programming language
2. Get knowledge about heterogeneous and homogeneous databases and concurrency control in distributed databases.
3. Explain the various types of locking mechanisms utilized within database management systems and database query processing.
4. Explain the different types of database failures as well as the methods used to recover from these failures.
5. Acquire knowledge about database tuning optimization and active and deductive databases.
6. Perform queries against database designed with object-relational extensions.
7. Understand the concept of special and multimedia databases.

**UNIT-I**

Object oriented databases - Complex data types, Object-oriented data model, Object-oriented languages, Persistent programming languages – Object relational databases - Nested relations, Complex types, Inheritance, Reference types, Querying with complex types, Functions and procedures, Object-oriented versus object-relational .

**UNIT-II**

Distributed databases - Homogeneous and heterogeneous databases, Distributed data storage, Distributed transactions, Commit protocols, Concurrency control in distributed databases, Availability, Distributed query processing, Heterogeneous distributed databases, Directory systems – Parallel databases - I/O parallelism, Inter query parallelism, Intra query parallelism, Intra operation parallelism, Inter operation parallelism, Design of parallel systems.

**UNIT-III**

Spatial databases and spatial, Geographic data - Representation of geometric information - Design databases, Geographic data, Spatial queries, Indexing of spatial data – Temporal and time series databases - Time in databases- Time specification in SQL, Temporal query language.

**UNIT-IV**

Multimedia databases – Multimedia data formats, Continuous media data, Similarity-based retrieval - Web databases – Web fundamentals, URL, HTML, Client side scripting and Applets, Web servers and sessions, Servlets, Server side scripting, Improving performance.

**UNIT-V**

Rules - Active and deductive databases - Security - Integrity - Consistency - Database Tuning optimization and Research issues.

**TEXT BOOK**

1. Henry Korth, F., Abraham Silberchatz, Sudarshan, S., Database System Concepts, 4<sup>th</sup> Edition , Mc Graw Hill International Editions.
2. Elmasri, R., Navathe, S.B., Fundamentals of Database Systems , Addison Wesley, 2000.

**REFERENCES**

1. Gary Hanson,W., James Hanson, V., Database Management and Design, Prentice Hall of India Pvt. Ltd., 1999.
2. Alex Benson, Stephen Smith and Kurt Thearling, Building Data Mining Applications for CRM, Tata McGraw-Hill, 2000.
3. Stefano Ceri, Giuseppe Pelagatti, Distributed Databases: Principles and Systems , Mc Graw-Hill Computer Science Series.

<b>BCS512</b>	<b>MOBILE COMPUTING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**COURSE OBJECTIVES**

- To understand the fundamentals and various computational processing of mobile networks.
- To study the specifications and functionalities of various protocols/standards of mobile networks

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Understand working, characteristics and limitations of mobile hardware devices including their user-interface modalities.
2. Understand and learn frequency band, spectrum, air interface and channel structure for telecommunication and identify the GSM, GPRS and Bluetooth software model for mobile computing.



3. Understand, analyze and explain problems associated to localization and movements and the wireless and wired communication architecture and handling of data and business application over slow wireless networks
4. Identify business data management and security issues over slow wireless media and Working of software mobile agents over long distances
5. Understand and learn transaction processing over wire and wireless media and various routing and communication protocols
6. Analyze QoS over wire and wireless channels and understand and recognize CDMA and other network applications

#### **UNIT-I**

Introduction - Applications, Reference model - Wireless transmission - Signal propagation, Multiplexing, Modulation, Spread spectrum.

#### **UNIT-II**

SDMA – FDMA – TDMA – Fixed TDMA, Classical Aloha, Slotted Aloha, CSMA, Demand assigned MA, PRMA, Reservation TDMA, MA with collision avoidance, Polling, ISMA– CDMA – Telecommunications system - GSM , DECT, TETRA, UMTS and IMT-2000.

#### **UNIT-III**

Basics of satellite communication - GEO, LEO, MEO – Broadcast System - Overview of broadcasting, Digital audio broadcasting, Digital video broadcasting.

#### **UNIT-IV**

IEEE 802.11 – Architecture, Protocol, MAC layer, Physical layer, 802.11b, 802.11a - HIPERLAN – HIPERLAN1, WATM, BRAN, HiperLAN2 – Bluetooth - Architecture, Radio layer, Base band layer, Link manager protocol, Security, SDP, Profiles.

#### **UNIT-V**

Mobile IP – Dynamic host configuration protocol – Mobile ad-hoc networks – WAP - Architecture, WAP, Wireless transport layer security, Wireless transaction protocol, Session protocol, Wireless markup language, WML script, Wireless telephony application, Push architecture, Push / Pull services.

#### **TEXT BOOK**

1. Jochen Schiller, Mobile Communication, 2<sup>nd</sup> Edition, Pearson Education, 2004.

#### **REFERNCE BOOK**

1. William Lee, C.Y., Mobile Communication Engineering, 2<sup>nd</sup> Edition, McGraw Hill.
2. Ashoke K Talukder, Roopa R Yavagal “Mobile Computing” Tata McGraw Hill 2005.

<b>BCS513</b>	<b>E-COMMERCE</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- To understand the interest and opportunity of e-commerce
- To know how to plan and how to manage e-commerce solutions
- To apply processes of e-commerce and to analyze and understand the human, technological and business environment associated with e-commerce
- To know how to use technologies to build e-commerce websites

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Define and analyze the principles of E-commerce and basics of World Wide Web.
2. Define and analyze the concept of electronic data interchange and its legal, social and technical aspects.
3. Define and analyze the security issues over the web, the available solutions and future aspects of e-commerce security.
4. Define and analyze the concept of E-banking, electronic payment system

**UNIT-I**

The difference between e-commerce and e-business - Types of e-commerce - Origins and Growth of E-commerce - Impact of E-commerce on Consumer, Organization and Society - Benefits and Limitations of E-commerce to Consumer, Organization and Society - Barriers to E-commerce - E-commerce business models and concepts

**UNIT-II**

Planning - Systems analysis and design - Building the system: In-house vs. outsourcing - Website hosting: In-house vs. outsourcing - System Testing - Implementation and maintenance - Website optimization factors - Choosing web server software - Logistics and Order Fulfilment - Overview of E-commerce website development tools

**UNIT-III**

Payment systems - Merchant Accounts, Payment Gateways, Credit Cards Processing & Third Party Payment Processors - E-commerce digital payment in the B2C arena-Electronic Check, E-Cash, SET based payment systems

**UNIT-IV**

The Internet audience - Internet traffic patterns - Basic marketing concepts - On-site Marketing Techniques – customer feedback, links, banner ads, affiliate programs, newsletters, emails

## **UNIT-V**

Impact of E-commerce on society - Legal/policy issues in e-commerce - Auctions- Portals - Communities

### **TEXT BOOKS**

1. Frontiers of E Commerce , Ravi Kalakota, Andrew B Whinston, Pearson Education(2002)

### **REFERENCES**

1. The Complete E-Commerce Book, Second Edition: Design, Build & Maintain a Successful Web-based Business by Janice Reynolds
2. E-commerce: Get It Right! By Ian Daniel
3. E-Commerce 2009: Business, Technology, and Society by Kenneth Laudon and Carol Guercio Traver

**SIXTH SEMESTER ELECTIVES**

<b>BCS611</b>	<b>CLOUD COMPUTING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- This course introduces a series of current cloud computing technologies, including technologies for Infrastructure as a Service, Platform as a Service, Software as a Service, and Physical Systems as a Service. Objective of this course is to learn different layers of the cloud technologies, practical solutions such as Google, Amazon, Microsoft, SalesForce.com, etc. solutions as well as theoretical solutions.

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Define cloud computing and related concepts
2. Understand the key dimensions of the challenges of Cloud Computing
3. Understand the assessment of the economics , financial, and technological implications for selecting cloud computing for an organization
4. Describe the benefits of cloud computing and to understand different layers of the cloud technologies, practical solutions
5. Understand the challenges of cloud computing
6. Understand how cloud components fit together
7. Determine the suitability of in-house v/s hosted solutions

**UNIT I**

Defining Cloud Computing: Definition - Cloud Types - Characteristics of Cloud Computing - Role of Open standards - Cloud Architecture: Cloud Computing Stack: Composibility.

**UNIT II**

Infrastructure - Platforms - Virtual Appliances - Communication protocols - Applications - Connecting to the cloud - Cloud Services: Infrastructure as a Service - Platform as a Service - Software as a Service

**UNIT III**

Identity as a Service - Compliance as a Service - Platforms: Load balancing and visualization– Understanding Hypervisors - Cloud Security: Securing the Cloud.

**UNIT IV**

Securing the data - Moving applications to the cloud - Cloud Storage: Definition – Provisioning –Cloud storage - Cloud Backup solutions - Cloud storage Interoperability

**UNIT V**

Moving applications to the Cloud - Case Study: Google Web Services- Amazon Web Services - Microsoft Cloud Services.

**TEXT BOOK:**

Barrie Sosinsky- Cloud Computing Bible- Wiley India Pvt. Ltd.- 2011.

**REFERENCE BOOKS:**

1. Roger Jennings- Cloud Computing with Windows Azure Platform- Wiley India Pvt. Ltd- 2009.
2. Miller Michael- Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online- Que Publishing- 2008.

<b>BCS612</b>	<b>DATA MINING AND DATA WAREHOUSING</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- To interpret the contribution of data warehousing and data mining to the decision support level of organizations
- To evaluate different models used for OLAP and data pre-processing
- To categorize and carefully differentiate between situations for applying different data mining techniques: mining frequent pattern, association, correlation, classification, prediction, and cluster analysis

**COURSE OUTCOMES**

Upon successful completion of this course, students will be able to

1. Understand the data extraction and transformation techniques.
2. List the association rule mining techniques and understand association mining to correlation analysis, constraint based association mining.
3. Understand operational database, warehousing and multidimensional need of data base to meet industrial needs.
4. Understand the components of warehousing, classification methods and clustering analysis.
5. Identify and understand the Business analysis, query tools and application, OLAP etc.

### **UNIT I**

Data Preprocessing- Language- Architectures- Concept Description: Preprocessing- Cleaning- Integration- Transformation- Reduction- Discretization- Concept Hierarchy Generation- Data Mining Primitives- Query Language- Graphical User Interfaces- Architectures- Concept Description- Data Generalization- Characterizations- Class Comparisons- Descriptive Statistical Measures.

### **UNIT II**

Association Rule: Association Rule Mining- Single-Dimensional Boolean Association Rules from Transactional Databases- Multi-Level Association Rules from Transaction Databases- mining multidimensional Association rules –association mining to correlation analysis- constraint based association mining.

### **UNIT III**

Classification and Prediction: Classification and Prediction- Issues- Decision Tree Induction- Bayesian Classification- Association Rule Based- Other Classification Methods- Prediction- Classifier Accuracy.

### **UNIT IV**

Cluster Analysis: Cluster Analysis- Types of data- Categorization of methods- Partitioning methods- hierarchical methods- density based methods- grid based methods - Outlier Analysis. Recent trends - Multidimensional Analysis and Descriptive Mining of Complex Data Objects- Spatial Databases- Multimedia Databases- Time Series and Sequence Data- Text Databases- World Wide Web- Applications and Trends in Data Mining

### **UNIT V**

Data Warehousing: Introduction- Data Warehouse- Multidimensional Data Model- Data Warehouse Architecture- Implementation - Data Warehousing to Data Mining -Data warehousing components-building a data warehouse – mapping the data warehouse to an architecture - data extraction - cleanup- transformation tools- metadata – OLAP - Patterns and models - Data visualization principles.

### **TEXT BOOKS**

1. J. Han and M. Kamber- “Data Mining: Concepts and Techniques”- Harcourt India Morgan Kauffman- 2001.
2. Alex Berson and Stephen J. Smith- “Data Warehousing- Data mining and OLAP”- Tata McGraw-Hill- 2004.

### **REFERENCES**

1. Margaret H. Dunham- “Data Mining: Introductory and Advanced Topics”- Pearson Education- 2004.
2. Sam Anahory and Dennis Murry- “Data Warehousing in the Real World”- Pearson Education- 2003

<b>BCS613</b>	<b>INFORMATON SECURITY</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
		3	0	0	3

**COURSE OBJECTIVES**

- To select appropriate techniques to tackle and solve problems in the discipline of information security management;
- To know why security and its management are important for any modern organisation;

**COURSE OUTCOMES**

Upon successful completion of this course, the students would be able to

1. Gain comprehensive information about security policies, establishing necessary organizational processes /functions for information security and will be able to arrange necessary resources.
2. Explain web security threats and SSL architecture
3. Gain knowledge about Symmetric Encryption Principles and algorithms
4. Know the hash functions and public key cryptography principles
5. Identify the threats to information security and Show how to protect information recourses
6. Show how to maintaining and protecting information system
7. Understand malicious software and have knowledge of cyber law and ethics.

**UNIT-I**

Information Security - Security Goals - Security Attacks - Security Services – Security mechanisms - A model for Internetwork security.

**UNIT-II**

Crypto Basics - Classic Cryptography - Symmetric Key Cryptography: Stream Ciphers - RC4, Block Ciphers - Feistel Cipher, DES, Triple DES, AES - Public Key Cryptography: RSA, Defiie-Hellman, uses of public key cryptography - Public Key Infrastructure, Hash Function : The Birthday problem – HMAC - Uses of hash functions.

**UNIT-III**

Authentication Methods – Passwords – Biometric – Two-factor authentication - Access control Matrix, ACLs, Multiple level security models, Multilateral security.

**UNIT-IV**

Software Flaws - Buffer Overflow, Incomplete Mediation, Race conditions, Malware - Salami attack, Linearization Attacks, Trusting Software, Software reverse engineering, Digital Rights management.

**UNIT-V**

Network security basics, TCP/IP Model and Port No., Protocol flaws, Enterprise wide network -Design and Vulnerabilities, Reconnaissance of network, Packet sniffing, Session Hijacking, -ARP Spoofing, Web site and web server vulnerabilities, Denial of Service, SSL and IPSec protocol, Firewall, Intrusion Detection System, and Honey pots, Email Security.

### **TEXT BOOK**

1. Mark Stamp, Information Security Principles & Practice, WILEY INDIA 2006.
2. Charles P. Pfleeger, "Security in Computing", Pearson Education

### **REFERENCES**

1. Behrouz A. Forouzan, Cryptography & Network Security, TMH 2007.
2. Robert Bragg, Mark Rhodes, Network Security: The complete reference, TMH
3. Wenbo Mao, Modern Cryptography, Pearson Education 2007.
4. William Stallings, Network Security Essentials: Applications and Standards, Fourth edition, PHI.



