BANKURA UNIVERSITY

Curriculum and Credit Framework for

BCA Programme

(Basic, Honours and Honours with Research)

designed in accordance with

Learning Outcomes-Based Curriculum Framework (LOCF)
of National Education Policy (NEP)-2020

SYLLABUS

With Effect from the Session 2023-2024

BANKURA UNIVERSITY

Curriculum and Credit Framework for

BCA Programme

(Basic, Honours and Honours with Research)

Preamble:

Education is the key to development of any society. Role of higher education is crucial for securing right kind of employment and also to pursue further studies in best available world class institutes elsewhere within and outside India. Quality education in general and higher education in particular deserves high priority to enable the young and future generation of students to acquire skill, training and knowledge in order to enhance their thinking, creativity, comprehension and application abilities and prepare them to compete, succeed and excel globally. Sustained initiatives are required to reform the present higher education system for improving and upgrading the academic resources and learning environments by raising the quality of teaching and standards of achievements in learning outcomes in undergraduate program in professional streams of higher education like computer science. One of the significant reforms in the undergraduate education is to introduce the Learning Outcomes-based Curriculum Framework (LOCF) which makes it student- centric, interactive and outcome-oriented with well-defined aims, objectives and goals to achieve. LOCF also aims at ensuring uniform education standard and content delivery across the country which will help the students to ensure similar quality of education irrespective of the institute and location. With initiatives of University Grants Commission (UGC) for nation-wide adoption and implementation of the LOCF for bachelor's programmes in colleges, universities and HEIs in general. So BCA Programme at Bankura University is designed as per LOCF & and NEP-2020 as per UGC guidelines.

The main objective of BCA Programme program at Bankura University is to prepare a comprehensive course structure with detailed syllabus along with quality reading material in order to have a uniform standard of education in BCA Programme among students. This document shall serve as a model document across the higher education institutes (HEIs) in the country for teachers, students and academic administrators. It is a student centric framework where they are expected to learn fundamentals of computer science along with the latest trends and techniques like Artificial Intelligence, .NET Technologies, Digital Image Processing, Data Warehousing and Data Mining along with advanced skillsets that include Mobile Application Development, Object Oriented Programming among many other courses. It will help the students to be equipped with fundamental as well as advanced and latest technologies in computer science after completion of the programme

Introduction:

BCA Programme has been evolving as an important branch of science and engineering throughout the world in last couple of decades and it has carved out a space for itself like any other disciplines of basic science and engineering. BCA Programme is a discipline that spans theory and practice and it requires thinking both in abstract terms and in concrete terms. Nowadays, practically everyone is a computer user, and many people are even computer programmers. BCA Programme can be seen on a higher level, as a science of problem solving and problem solving requires precision, creativity, and careful reasoning. The ever-evolving discipline of computer science also has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers, but finding a solution requires both computer science expertise and knowledge of the particular application domain.

BCA Programme has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Computational Science, and Software Engineering. Drawing from a common core of computer science knowledge, each specialty area focuses on specific challenges. BCA Programme is practised by mathematicians, scientists and engineers. Mathematics, the origins of Computer Science, provides reason and logic. Science provides the methodology for learning and refinement. Engineering provides the techniques for building hardware and software.

BANKURA UNIVERSITY

Bankura University has started BCA Programme in the year 2018, as this discipline evolved itself to a multidisciplinary discipline. Information Technology is growing rapidly. Increasing applications of computers in almost all areas of human endeavor has led to vibrant industries with concurrent rapid change in technology. Unlike other basic disciplines, developing core competency in this discipline that can be reasonably stable becomes a challenge.

Career Objective:

BCA Programme are aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in Computer Science/Computer Application leading to research as well as R&D, can be employable at IT industries, or can pursue a teachers' training programme such B. Ed. in Computer Application, or can adopt a business management career. BCA Programme aims at laying a strong foundation of Computer Application at an early stage of the career. There are several employment opportunities and after successful completion of an undergraduate programme in BCA Programme, graduating students can fetch employment directly in companies as Web Developer, Software Engineer, Network Administrator, Data Scientist, or AI/ML personnel.

The Learning Outcome-based Curriculum Framework in BCA Programme is aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teachinglearning process and in assessment procedures of the learning outcomes. The emphasis in BCA Programme course, in outcome-based curriculum framework, help students learn solving problems, accomplishing IT tasks, and expressing creativity, both individually and collaboratively. The proposed framework will help Students learn programming techniques and the syntax of one or more programming languages.

Many of the learning outcomes of BCA Programme can be achieved only by programming a computer for several different meaningful purposes. All students must, therefore, have access to a computer with a modern programming language installed. The computer science framework does not prescribe a specific language. The teacher and students will decide which modern programming languages students will learn. More importantly, students will learn to adapt to changes in programming languages and learn new languages as they are developed.

- The present Learning Outcome-based Curriculum Framework for BCA Programme is intended to facilitate the students to achieve the following.
- To develop an understanding and knowledge of the basic theory of Computer Science and Information Technology with good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation.
- To develop the ability to use this knowledge to analyse new situations.
- To acquire necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit to the above-mentioned purpose.
- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems.
- To learn skills and tools like mathematics, statistics, physics and electronics to find the solution, interpret the results and make predictions for the future developments.

BANKURA UNIVERSITY

Curriculum Planning- Learning Outcomes-based Approach for BCA Programme:

BCA Programme is generally a three-year degree program which develops advanced theoretical and research skills in subject. This programme helps in building an advanced professional or academic career. It is an appropriate course for students who wish to pursue a MCA, M. Sc. (Computer Science), MBA or Doctor of Philosophy (PhD) in CS and a research or academic career. This program facilitates students who wish to pursue an independent research project in an area of interest under the supervision of an academic

Aims of BCA Programme:

The BCA Programme emphasizes problem solving in the context of algorithm development and software implementation and prepares students for effectively using modern computer systems in various applications. The curriculum provides required BCA Programme courses such as programming languages, data structures, computer architecture and organization, algorithms, database systems, operating systems, and software engineering; as well as elective courses in artificial intelligence, computer-based communication networks, distributed computing, information security, graphics, human-computer interaction, multimedia, scientific computing, web technology, and other current topics in computer science. The main aim of this Bachelor's degree is to deliver a modern curriculum that will equip graduates with strong theoretical and practical backgrounds to enable them to excel in the workplace and to be lifelong learners.

The purpose of the BCA Programme programs in are twofold:

(1) to prepare the student for a position involving the design, development and implementation of computer software/hardware, and

(2) to prepare the student for entry into a program of postgraduate study in computer science/engineering and related fields.

BCA Programme focus on the concepts and techniques used in the design and development of software systems. Students in this program explore the conceptual underpinnings of Computer Science -- its fundamental algorithms, programming languages, operating systems, and software engineering techniques. In addition, students choose from a rich set of electives that includes data science, computer graphics, artificial intelligence, database systems, computer architecture, and computer networks, among other topics. A generous allotment of free electives allows students to combine study in computer science with study in auxiliary fields to formulate a program that combines experiences across disciplines.

Programme Learning Outcomes for BCA Programme:

The BCA Programme program enables students to attain, by the time of graduation:

- **PLO-A:** Demonstrate the aptitude of Computer Programming and Computer based problem solving skills.
- **PLO-B:** Display the knowledge of appropriate theory, practices and tools for the specification, design, implementation
- **PLO-C:** Ability to learn and acquire knowledge through online courses available at different MOOC Providers.
- **PLO-D:** Ability to link knowledge of Computer Science with other two chosen auxiliary disciplines of study.
- **PLO-E:** Display ethical code of conduct in usage of Internet and Cyber systems.
- **PLO-F:** Ability to pursue higher studies of specialization and to take up technical employment.
- **PLO-G:** Ability to formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate.

BANKURA UNIVERSITY

- **PLO-H:** Ability to operate, manage, deploy, configure computer network, hardware, software operation of an organization.
- **PLO-I:** Ability to present result using different presentation tools.
- **PLO-J:** Ability to appreciate emerging technologies and tools.
- **PLO-K:** Apply standard Software Engineering practices and strategies in real-time software project development
- **PLO-L:** Design and develop computer programs/computer -based systems in the areas related to algorithms, networking, web design, .NET Technology and data analytics.
- **PLO-M:** Acquaint with the contemporary trends in industrial/research settings and thereby innovate novel solutions to existing problems
- **PLO-N:** The ability to apply the knowledge and understanding noted above to the analysis of a given information handling problem.
- **PLO-O:** The ability to work independently on a substantial software project and as an effective team member.

BANKURA UNIVERSITY

Curriculum and Credit Framework for

BCA Programme

(Basic, Honours and Honours with Research)

With effect from the Academic Year : 2023-2024

SEMESTER – I

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/101/MJC-1	Computer Fundamentals	4	20	80		100	3	1	0
					T:80	L:00				
2	BCA/102/MN-1	Elective-I	4	See in POOL OF MINOR STREAM (ELECTIVE)						
3	BCA/103/MD-1	Mathematics-I	3	20	80		100	3	1	0
					T:80	L:00				
4	BCA/104/SEC-1	PC Software	3	20	80		100	3	1	4
					T:50	L:30				
5	BCA/105/AEC-1	Compulsory English: Literature and Communication	2	10	40		50	2	1	0
					T:40	L:00				
6	BCA/106/VAC-1	Environmental Studies	4	10	40		50	2	1	0
					T:40	L:00				
Total in Semester-I			**20**	**100**	**400**		**500**			

SEMESTER – II

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/201/MJC-2	Programming and Problem Solving through C-Language	4	20	80		100	3	1	4
					T:50	L:30				
2	BCA/202/MN-2	Elective-II	4	See in POOL OF MINOR STREAM (ELECTIVE)						
3	BCA/203/MD-2	Mathematics-II	3	20	80		100	3	1	0
					T:80	L:00				
4	BCA/204/SEC-2	Front End Web Design and Development	3	20	80		100	3	1	4
					T:50	L:30				
5	BCA/205/AEC-2 (A/B/C)	Modern Indian Language-I **Choose any one from the following** • A. Santali / B. Sanskrit / C. Bengali	2	10	40		50	2	1	0
					T:40	L:00				
6	BCA/206/VAC-2 (A/B/C/D/E)	**Choose any one from the following** • A. Health and Wellness • B. Understanding India: Indian Philosophical Traditions and Value Systems • C. Basics of Indian Constitution • D. Arts and Crafts of Bengal • E. Historical Tourism in West Bengal	4	10	40 T:40	L:00	50	2	1	0
Total in Semester-II			**20**	**100**	**400**		**500**			
7	BCA/207/INT-1	Summer Internship (Additional for getting UG CERTIFICATE)	**4**	**20**	80		100	0	0	4
					T:00	L:80				

UG CERTIFICATE	Student who opt to exit after completion of the first year (Semester-I & II) and have secure 40 credits will be awarded a **UG CERTIFICATE** if, in addition, the Summer Internship (4 Credit) (Sl. No.-7) during summer vacation of the first year.

SEMESTER – III

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/301/MJC-3	Digital Logic	4	20	80 T:50	L:30	100	3	1	4
2	BCA/302/MJC-4	Object Oriented Programming using C++	4	20	80 T:50	L:30	100	3	1	4
3	BCA/303/MN-3	Elective-III	4	See in POOL OF MINOR STREAM (ELECTIVE)						
4	BCA/304/MD-3	Mathematics-III	3	20	80 T:80	L:00	100	3	1	0
5	BCA/305/SEC-3	Digital Marketing	3	20	80 T:80	L:00	100	3	1	0
6	BCA/306/AEC-3 (A/B/C)	Modern Indian Language-II **Choose any one from the following** A. Bengali / B. Sanskrit / C. Santali	2	10	40 T:40	L:00	50	2	1	0
Total in Semester-III			**20**	**110**	**440**		**550**			

SEMESTER – IV

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/401/MJC-5	Operating System	4	20	80 T:50	L:30	100	3	1	4
2	BCA/402/MJC-6	Software Engineering	4	20	80 T:80	L:00	100	3	1	0
3	BCA/403/MJC-7	Data Structure & Algorithm through C++	4	20	80 T:50	L:30	100	3	1	4
4	BCA/404/MJC-8	Computer Networks	4	20	80 T:50	L:30	100	3	1	4
5	BCA/405/MN-4	Elective-IV	4	See in POOL OF MINOR STREAM (ELECTIVE)						
6	BCA/406/AEC-4	Compulsory English : Literature Language and Communication	2	10	40 T:40	L:00	50	2	1	0
Total in Semester-IV			**22**	**110**	**440**		**550**			
7	BCA/407/INT-2	Summer Internship (Additional for getting UG DIPLOMA)	**4**	**20**	80 T:00	L:80	100	0	0	4
UG DIPLOMA		Student who opt to exit after completion of the second year (Semester-I, II, III & IV) and have secure 82 credits will be awarded a **UG DIPLOMA** if, in addition, the Summer Internship (Sl. No.-7) (4 Credit) during summer vacation of the second year.								

SEMESTER – V

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/501/MJC-9	Computer Graphics and Multimedia	4	20	80		100	3	1	4
					T:50	L:30				
2	BCA/502/MJC-10	Database Management Systems	4	20	80		100	3	1	0
					T:50	L:30				
3	BCA/503/MJC-11	Microprocessor and System Software	4	20	80		100	3	1	4
					T:50	L:30				
4	BCA/504/MJC-12	Introduction to Internet of Things	4	20	80		100	3	1	0
					T:80	L:00				
5	BCA/505/MN-5	Elective-V	4	See in POOL OF MINOR STREAM (ELECTIVE)						
6	BCA/506/INT-3	Major Project & Viva-Voce	2	20	80		100	3	1	4
					T:00	L:80				
Total in Semester-V			**22**	**120**	**480**		**600**			

SEMESTER – VI

Sl. No.	Course Code	Course Title	Credit	Marks			No. of Hours		
				IA	ESE	Total	L	T	P
1	BCA/601/MJC-13	Programming in JAVA	4	20	80	100	3	1	4
					T:50 L:30				
2	BCA/602/MJC-14	Internet Systems	4	20	80	100	3	1	4
					T:50 L:30				
3	BCA/603/MJC-15	.NET Technologies	4	20	80	100	3	1	4
					T:50 L:30				
4	BCA/604/MJC-16	Object Oriented Database Management System	4	20	80	100	3	1	0
					T:80 L:00				
5	BCA/605/MN-6	Elective-VI	4	See in POOL OF MINOR STREAM (ELECTIVE)					
Total in Semester-VI			**20**	**100**	**400**	**500**			

UG DEGREE (BCA)	Student who wish to undergo three year UG Programme will be awarded **UG DEGREE (BCA)** after successful completion of three year (Semester-I, II, III, IV, V & VI).

SEMESTER – VII

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/701/MJC-17	Theory of Computation	4	20	80		100	3	1	0
					T:80	L:00				
2	BCA/702/MJC-18	Python Programming	4	20	80		100	3	1	4
					T:50	L:30				
3	BCA/703/MJC-19	Digital Image Processing	4	20	80		100	3	1	0
					T:80	L:00				
4	BCA/704/MJC-20	Cloud Computing	4	20	80		100	3	1	0
					T:80	L:00				
5	BCA/705/MN-7	Elective-VII	4	See in POOL OF MINOR STREAM (ELECTIVE)						
Total in Semester-VII			**20**	**100**	**400**		**500**			

SEMESTER – VIII

Sl. No.	Course Code	Course Title	Credit	Marks				No. of Hours		
				IA	ESE		Total	L	T	P
1	BCA/801/MJC-21	Distributed Systems	4	20	80		100	3	1	0
					T:80	L:00				
2	BCA/802/MJC-22	Unix and Shell Programming	4	20	80		100	3	1	4
					T:50	L:30				
3	BCA/803/MJC-23	Database Programming with PL/SQL	4	20	80		100	3	1	4
					T:50	L:30				
4	BCA/804/MJC-24	Web Technologies	4	20	80		100	3	1	0
					T:80	L:00				
5	BCA/805/MN-8	Elective-VIII	4	See in POOL OF MINOR STREAM (ELECTIVE)						
Total in Semester-VIII			**20**	**100**	**400**		**500**			

UG DEGREE(Hons.) (BCA (Hons.))	Student who wish to undergo four year UG Honours Programme will be awarded **UG DEGREE (HONOURS.) (BCA (Hons.))** after successful completion of four year (Semester-I, II, III, IV, V, VI, VII & VIII).
UG DEGREE (Hons. with Research) (BCA (Hons. with Research))	Students who secure 75% marks and above first six semesters and wish to undertake research at the undergraduate level can choose a research stream in the fourth year in replace of BCA/802/MJC-22, BCA/803/MJC-23 and BCA/804/MJC-24 papers of Semester-VIII. The students who secure 164 credits, including 12 credits from a research project/dissertation, are awarded **UG DEGREE (HONOURS WITH RESEARCH) (BCA (Hons. with Research)**
RESEARCH STREAM	

Sl. No.	Course Code	Course Title	Credit	Marks			No. of Hours		
				IA	ESE	Total	L	T	P
2	BCA/802/RPD-1	Research Methodology and IPR	4	20	80 (T:80 | L:00)	100	3	1	0
3	BCA/803/RPD-2	Research and Publication Ethics	4	20	80 (T:80 | L:00)	100	3	1	0
4	BCA/804/RPD-3	Research Project/Dissertation and Viva-voce	4	20	80 (T:00 | L:80)	100	3	1	4

Note: ***Besides the normal Curriculum, one Mandatory Non-Credit Course (MNC) to be introduced for each year. For the MNC course, weekly one class (either online of offline mode) may be arranged and the evaluation scheme may be entirely left over the Faculty member handling the course. Students may also be permitted to register for these courses in SWAYAM and NPTEL. These courses do not carry any credits.***

POOL OF MINOR STREAM (ELECTIVE)

A. Artificial Intelligence (AI) and Machine Learning (ML)

Course Code	Course Title	Marks			No. of Hours		
		IA	ESE	Total	L	T	P
MN-1A	Introduction to Artificial Intelligence	20	80 (T:80 | L:00)	100	3	1	0
MN-2A	Introduction to Machine Learning	20	80 (T:80 | L:00)	100	3	1	0
MN-3A	Deep Learning	20	80 (T:80 | L:00)	100	3	1	0
MN-4A	Fuzzy Logic and its Applications	20	80 (T:80 | L:00)	100	3	1	0
MN-5A	Knowledge Engineering and Expert Systems	20	80 (T:80 | L:00)	100	3	1	0
MN-6A	LISP Programming	20	80 (T:50 | L:30)	100	3	1	4
MN-7A	Neural Network	20	80 (T:80 | L:00)	100	3	1	0
MN-8A	Introduction to Natural Language Processing (NPL)	20	80 (T:80 | L:00)	100	3	1	0

B. Data Science (DS)

Course Code	Course Title	Marks			No. of Hours		
		IA	ESE	Total	L	T	P
MN-1B	Introduction to Data Science	20	80 (T:80 | L:00)	100	3	1	0
MN-2B	Data Security and Privacy	20	80 (T:80 | L:00)	100	3	1	0
MN-3B	Data Warehousing and Data Mining	20	80 (T:80 | L:00)	100	3	1	0
MN-4B	Data Modeling	20	80 (T:80 | L:00)	100	3	1	0
MN-5B	Statistics for Data Science	20	80 (T:80 | L:00)	100	3	1	0
MN-6B	Business Intelligence	20	80 (T:80 | L:00)	100	3	1	0
MN-7B	R – Programming for Data Science	20	80 (T:50 | L:30)	100	3	1	4
MN-8B	Big Data Analytics and Visualization	20	80 (T:80 | L:00)	100	3	1	0

C. Cyber Securities

Course Code	Course Title	Marks			No. of Hours		
		IA	ESE	Total	L	T	P
MN-1C	Introduction to Cyber Securities	20	80 (T:80 | L:00)	100	3	1	0
MN-2C	Cyber Laws in India	20	80 (T:80 | L:00)	100	3	1	0
MN-3C	Indian IT Act	20	80 (T:80 | L:00)	100	3	1	0
MN-4C	Fundamental Security Design Principles	20	80 (T:80 | L:00)	100	3	1	0
MN-5C	Network Security	20	80 (T:80 | L:00)	100	3	1	0
MN-6C	Cryptography and Data Security	20	80 (T:80 | L:00)	100	3	1	0
MN-7C	Legal & Ethical Issues and Ethical Hacking	20	80 (T:80 | L:00)	100	3	1	0
MN-8C	Cyber Forensics	20	80 (T:80 | L:00)	100	3	1	0

D. Software Engineering

Course Code	Course Title	Marks			No. of Hours		
		IA	ESE	Total	L	T	P
MN-1D	Principles of Management	20	80 (T:80 | L:00)	100	3	1	0
MN-2D	System Analysis and Design	20	80 (T:80 | L:00)	100	3	1	0
MN-3D	IT Service Management	20	80 (T:80 | L:00)	100	3	1	0
MN-4D	Software Designing	20	80 (T:80 | L:00)	100	3	1	0
MN-5D	Advance Software Engineering	20	80 (T:80 | L:00)	100	3	1	0
MN-6D	Software Project Management	20	80 (T:80 | L:00)	100	3	1	0
MN-7D	Software Testing	20	80 (T:80 | L:00)	100	3	1	0
MN-8D	Software Quality Assurance	20	80 (T:80 | L:00)	100	3	1	0

Note:

BCA = Bachelor of Computer Application, AEC = Ability Enhancement Course, MN=Minor Stream, MD=Multidisciplinary Course, SEC = Skill Enhancement Course, MJC= Major Discipline Specific Course, VAC=Value Added Course, INT = Internship, RPD = Research Project/ Dissertation, IA = Internal Assessment, ESE= End- Semester Examination, L = Lecture, T = Tutorial, and P = Practical

BANKURA UNIVERSITY

Curriculum and Credit Framework for

BCA Programme

(Basic, Honours and Honours with Research)

With effect from the Academic Year : 2023-2024

QUESTION PATTERN

Theory Paper:

Full Marks: 80

Group	No. of Question Present in Question Paper	No. of Question Attempt by the Student	Marks of Each Question	Total marks
A	10 (MCQ)	10	1	10
B	15	10	2	20
C	6	4	5	20
D	6	3	10	30
Total				80

Full Marks: 50

Group	No. of Question Present in Question Paper	No. of Question Attempt by the Student	Marks of Each Question	Total marks
A	10 (MCQ)	10	1	10
B	8	5	2	10
C	6	4	5	20
D	2	1	10	10
Total				50

Practical Paper:

Full Marks: 30

Group	No. of Question Present in Question Paper	No. of Question Attempt by the Student by Lottery	Marks of Each Question	Total marks
A	5	1	15	15
B	5	1	15	15
Total				30

Major Project & Viva-Voce:

Full Marks: 80

Particulars	Total marks
Project Report & Presentation	50
Viva-Voce	30
Total	80

Internal Assessment (IA):

Full Marks: 20 (Theory Paper)

Particulars	Marks
Class Attendance	5
Assignments	5
Seminars	5
Co-curricular Activities and Behavior	5
Total	20

Full Marks: 20 (Theory + Practical Paper)

Particulars	Marks
Class Attendance + Lab. Attendance	5
Assignments + Lab. Note Book	5
Seminars	5
Co-curricular activities and behavior	5
Total	20

BANKURA UNIVERSITY

Curriculum and Credit Framework for

BCA Programme

(Basic, Honours and Honours with Research)

With effect from the Academic Year : 2023-2024

LIST OF THE COURSES		
Sl. No.	**Course Code**	**Course Title**
1. ABILITY ENHANCEMENT COURSE (AEC):		
1	AEC-1	Compulsory English: Literature and Communication
2	AEC-2A	Modern Indian Language-I – Bengali
3	AEC-2B	Modern Indian Language-I – Sanskrit
4	AEC-2C	Modern Indian Language-I – Santali
5	AEC-3A	Modern Indian Language-II – Bengali
6	AEC-3B	Modern Indian Language-II – Sanskrit
7	AEC-3C	Modern Indian Language-II – Santali
8	AEC-4	Compulsory English : Literature Language and Communication
2. MAJOR STREAM:		
9	MJC-1	Computer Fundamentals
10	MJC -2	Programming and Problem Solving using C-Language
11	MJC -3	Digital Logic
12	MJC -4	Object Oriented Programming using C++
13	MJC-5	Operating System
14	MJC -6	Software Engineering
15	MJC -7	Data Structure & Algorithm through C++
16	MJC -8	Computer Networks
17	MJC -9	Computer Graphics and Multimedia
18	MJC -10	Database Management Systems
19	MJC -11	Microprocessor and System Software
20	MJC -12	Introduction to Internet of Things (IOT)
21	MJC -13	Programming in Java
22	MJC -14	Internet Systems
23	MJC -15	.NET Technologies
24	MJC-16	Object Oriented Database Management System
25	MJC -17	Theory of Computation
26	MJC -18	Python Programming
27	MJC -19	Digital Image Processing
28	MJC -20	Cloud Computing
29	MJC -21	Distributed Systems
30	MJC -22	Unix and Shell Programming
31	MJC -22	Database Programming with PL/SQL
32	MJC -24	Web Technologies

3. MINOR STREAM:		
A. Artificial Intelligence (AI) and Machine Learning (ML)		
33	MN-1A	Introduction to Artificial Intelligence
34	MN-2A	Introduction to Machine Learning
35	MN-3A	Deep Learning
36	MN-4A	Fuzzy Logic and its Applications
37	MN-5A	Knowledge Engineering and Expert Systems
38	MN-6A	LISP Programming
39	MN-7A	Neural Network
40	MN-8A	Introduction to Natural Language Processing (NPL)
B. Data Science (DS)		
41	MN-1B	Introduction to Data Science
42	MN-2B	Data Security and Privacy
43	MN-3B	Data Warehousing and Data Mining
44	MN-4B	Data Modeling
45	MN-5B	Statistics for Data Science
46	MN-6B	Business Intelligence
47	MN-7B	R – Programming for Data Science
48	MN-8B	Big Data Analytics and Visualization
C. Cyber Securities		
49	MN-1C	Introduction to Cyber Securities
50	MN-2C	Cyber Laws in India
51	MN-3C	Indian IT Act
52	MN-4C	Fundamental Security Design Principles
53	MN-5C	Network Security
54	MN-6C	Cryptography and Data Security
55	MN-7C	Legal & Ethical Issues and Ethical Hacking
56	MN-8C	Cyber Forensics
D. Software Engineering		
57	MN-1D	Principles of Management
58	MN-2D	System Analysis and Design
59	MN-3D	IT Service Management
60	MN-4D	Software Designing
61	MN-5D	Advance Software Engineering
62	MN-6D	Software Project Management
63	MN-7D	Software Testing
64	MN-8D	Software Quality Assurance
4. MULTIDISCIPLINARY COURSE		
65	MD-1	Mathematics-I
66	MD-2	Mathematics-II
67	MD-3	Mathematics-III
5. SKILL ENHANCEMENT COURSE (SEC):		
68	SEC-1	PC Software
69	SEC-2	Front End Web Design and Development

70	SEC-3	Digital Marketing
6. VALUE ADDED COURSE (VAC):		
71	VAC-1	Environmental Studies
72	VAC-2A	Health and Wellness
73	VAC-2B	Understanding India: Indian Philosophical Traditions and Value Systems
74	VAC-2C	Basics of Indian Constitution
75	VAC-2D	Arts and Crafts of Bengal
76	VAC-2E	Historical Tourism in West Bengal
7. SUMMER INTERNSHIP/PROJECT & VIVA-VOCE		
77	INT-1	Summer Internship
78	INT-2	Summer Internship
79	INT-3	Major Project & Viva-Voce
8. RESEARCH STREAM		
80	RPD-1	Research Methodology and IPR
81	RPD-2	Research and Publication Ethics
82	RPD-3	Research Project/Dissertation and Viva-voce

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Semester - I
Course Code: BCA/101/MJC-1
Course Title: Computer Fundamentals (4 Credit)

Course Objectives:
The main objective of this course is to introduce the fundamentals of computing devices and reinforce computer vocabulary, particularly with respect to personal use of computer hardware and software, the Internet, networking and mobile computing. It focuses on such computer literacy that prepares students for life-long learning of computer concepts and skills. Students discover why computers are essential components in education, business and society in this course.

Learning Outcomes: After successfully completing this course, a student will be able to:

- Student will be able to identify the components of a personal computer system
- Student will be able to demonstrate mouse and keyboard functions
- Student will be able to demonstrate window and menu commands and how they are used
- Student will be able to demonstrate how to organize files and documents on a USB/hard drive
- Student will be able to compose, format and edit a word document

Theory: 60 Lectures

Unit-I: Introduction: Introduction to computers, Evolution of computer, characteristics and capabilities, Block Diagram of a Computer, Types of Computers: Analogue, Digital, Hybrid, General and Special Purpose Computers, Generation of Computers, Computer Systems: Micros, Minis & Main-frames, Limitations of Micro Computer.

Unit-II: Number systems: Decimal Number system, Binary number system, Octal & Hexadecimal number system, 1's&2's complement Codes: ASCH, EBCDI Codes, Gray code & BCD.

Unit-III: Logic Gates: AND, OR, NOT GATES and their Truth tables, NOR, NAND & XOR gates

Unit-IV: Input Devices: Categorizing Input Hardware, Keyboard, Direct Entry — Card Readers, Scanning Devices — O.M.R., Character Readers, Thumb Scanner, MICR, Smart Cards, Voice Input Devices, Pointing Devices — Mouse, Light Pen, Touch Screen.

Unit-V: Output Devices: Output Fundamentals, Hardcopy Output Devices, Impact Printers, Non-Impact Printers, Plotters, Computer output Microfilm/Microfiche (COM) systems, Softcopy Output Devices, Cathode Ray Tube, Flat Screen Technologies, Projectors, Speakers.

Unit-VI: Central Processing Unit: The Microprocessor, control unit, ALU, Registers, Buses, Main Memory, RAM,ROM.

Unit-VII: Storage Devices: Storage Fundamentals, Primary and Secondary Storage, Data Storage and Retrieval Methods — Sequential, Direct & Indexed Sequential, Tape Storage and Retrieval Methods Tape storage Devices, characteristics and limitations, Direct access Storage and Microcomputers – Hard Disks, Disk Cartridges, Direct AccessStorage Devices for large Computer systems, Mass storage systems and Optical Disks, CD ROM.

Unit-VIII: Computer Software & Languages: System Software: System software Vs. Application Software, Types of System Software, Introduction and Types of Operating Systems. Boot Loader, Diagnostic Programs, BIOS, Utility Programs.
Application Software: Microcomputer Software, Interacting with the System, Trends in PC software, Types of Application Software, Difference between Program and Packages.
Computer Languages: Definition, Generations of computer languages, Types of Languages, Language Processors: Assembler, Interpreter, Compiler, Linker and Loader.

Unit-IX: Problem Solving: Algorithms, Flow Charts, Decision Tables and Pseudo Code.

Text Books/Reference Books/Online Resources:

1. Computer Fundamentals - P. K. Sinha
2. Rajaraman V. -Fundamentals of computers, Prentics hall of India.
3. Digital circuit and designs -S. Salivahanan, S. Arivazhagan - Vikash Publications.

Semester - I
Course Code: BCA/102/MN-1A
Course Title: Introduction to Artificial Intelligence (4 Credit)

Course Objectives:
Introduction to AI is an introductory course in Artificial Intelligence. The goal is to acquire knowledge on intelligent systems and agents, formalization of knowledge, reasoning with and without uncertainty, machine learning and applications at a basic level.

Learning Outcomes:
After the completion of this course, the students will be able to:
Identify problems where artificial intelligence techniques are applicable. Apply selected basic AI techniques; judge applicability of more advanced techniques. Participate in the design of systems that act intelligently and learn from experience.

Theory: 60 Lectures

Unit-I: Introduction to Artificial Intelligence: Definition of AI, Future of Artificial Intelligence, Characteristics of Intelligent Agents, Typical Intelligent Agents.

Unit-II: AI Project Cycle: Definition, Flow chart (AI Project Cycle), Problem Scoping, Data Acquisition, Data Exploration, Modelling.

Unit-III: Problem Solving Methods: Problem solving Methods, Search Strategies, Uninformed and Informed Search, Local Search, Heuristics, Algorithms and Optimization Problems, Searching with Partial Observations, Constraint: Satisfaction Problems, Constraint Propagation, Backtracking Search, Game Playing, Optimal Decisions in Games, Alpha-Beta Pruning, Stochastic Games.

Unit-IV: Knowledge Representation: Knowledge Representation, First-Order Predicate Logic, Prolog Programming, Unification, Forward and Backward Chaining, Resolution, Ontological Engineering, Categories and Objects, Events, Mental Events and Mental Objects, Reasoning Systems for Categories, Reasoning with Default Information.

Unit-V: Software Agents: Architecture for Intelligent Agents, Agent communication, Negotiation and Bargaining, Argumentation among Agents, Trust and Reputation in Multi-agent systems.

Unit-VI: Artificial Intelligence Applications: Artificial Intelligence applications, Language Models, Information Retrieval, Information Extraction, Natural Language Processing, Machine Translation, Speech Recognition, Robotics, Hardware and Software for Robots, Planning and Perception.

Text Books/Reference Books/Online Resources:

1. DAN.W. Patterson, Introduction to A.I and Expert Systems – PHI, 2007.
2. N. J. Nilson, Principle of AI, Narosa
3. Rich & Knight, Artificial Intelligence – Tata McGraw Hill, 2nd edition, 1991
4. Russell & Norvig, Artificial Intelligence-A Modern Approach, LPE, Pearson Prentice Hall, 2nd edition, 2005.

Semester - I
Course Code: BCA/102/MN-1B
Course Title: Introduction to Data Science (4 Credit)

Course Objectives:
This course is intended to provide an introduction into the field of Data Science. Students will develop skills in appropriate technology and basic statistical methods by completing hands-on projects focused on real-world data and addresses the social consequences of data analysis and application.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Identify and describe the methods and techniques commonly used in data science.
- Demonstrate proficiency with the methods and techniques for obtaining, organizing, exploring, and analyzing data.
- Recognize how data analysis, inferential statistics, modeling, machine learning, and statistical computing can be utilized in an integrated capacity.
- Create and modify customizable tools for data analysis and visualization per the evaluation of characteristics of the data and the nature of the analysis.
- Demonstrate the ability to clean and prepare data for analysis and assemble data from a variety of sources

Theory: 60 Lectures

Unit – I: Introduction: Introduction to Data Science – Evolution of Data Science – Data Science Roles – Stages in a Data Science Project – Applications of Data Science in various fields – Data Security Issues.

Unit – II: Data Collection and Data Pre-Processing: Data Collection Strategies – Data Pre-Processing Overview – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization.

Unit – III: Exploratory Data Analytics: Descriptive Statistics – Mean, Standard Deviation, Skewness and Kurtosis – Box Plots – Pivot Table – Heat Map – Correlation Statistics – ANOVA.

Unit – IV: Model Development: Simple and Multiple Regression – Model Evaluation using Visualization – Residual Plot – Distribution Plot – Polynomial Regression and Pipelines – Measures for In-sample Evaluation – Prediction and Decision Making.

Unit – V: Model Evaluation: Generalization Error – Out-of-Sample Evaluation Metrics – Cross Validation – Over fitting – Under Fitting and Model Selection – Prediction by using Ridge Regression – Testing Multiple Parameters by using Grid Search.

Text Books/Reference Books/Online Resources:

1. Jojo Moolayil, "Smarter Decisions : The Intersection of IoT and Data Science", PACKT, 2016.
2. Cathy O'Neil and Rachel Schutt , "Doing Data Science", O'Reilly, 2015.
3. David Dietrich, Barry Heller, Beibei Yang, "Data Science and Big data Analytics", EMC 2013
4. Raj, Pethuru, "Handbook of Research on Cloud Infrastructures for Big Data Analytics", IGI Global

Semester - I
Course Code: BCA/102/MN-1C
Course Title: Introduction to Cyber Securities (4 Credit)

Course Objectives:
The objective of the courses to

- Understand the fundamental s of cyber security and cybercrimes.
- Understand the tools and methods in cybercrimes and understanding computer forensics.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the basic concepts of cyber security and cybercrimes.
- Understand the security policies and cyber laws.

Theory: 60 Lectures

Unit-I: Introduction to Cybercrime: Cybercrime- Definition and Origins of the Word Cybercrime and Information Security, Who are Cybercriminals? Classifications of Cybercrimes, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens. Cyber offenses: How Criminals Plan Them: How Criminals Plan the Attacks, Social Engineering, Cyberstalking, Cybercafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector, Cloud Computing.

Unit-II: Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops.

Unit-III: Tools and Methods Used in Cybercrime: Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan-horses and Backdoors, Steganography, DoS and DDoS At-tacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks. Phishing and Identity Theft: Introduction to Phishing, Identity Theft (ID Theft).

Unit-IV: Understanding Computer Forensics: Introduction, Digital Forensics Science, The Need for Computer Forensics, Cyber forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Setting up a Computer

Forensics Laboratory: Understanding the Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics.

Unit-V: Forensics and Social Networking Sites: The Security/Privacy Threats, Computer Forensics from Compliance Perspective, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing, Ant forensics.

Unit-VI: Introduction to Security Policies and Cyber Laws: Need for an Information Security Policy, Information Security Standards – ISO, Introducing Various Security Policies and Their Review Process, Introduction to Indian Cyber Law, Objective and Scope of the IT Act, 2000, Intellectual Property Issues, Overview of Intellectual Property Related Legislation in India, Patent, Copyright, Law Related to Semiconductor Layout and Design, Software License.

Text Books/Reference Books/Online Resources:

1. SunitBelapure and Nina Godbole, "Cyber Security: Understanding Cyber Crimes, Computer Forensics And Legal Perspectives", Wiley India Pvt Ltd, ISBN: 978-81- 265-21791, Publish Date 2013.
2. Dr. Surya PrakashTripathi, RitendraGoyal, Praveen Kumar Shukla, KLSI. "Introduction to information security and cyber laws". Dreamtech Press. ISBN: 9789351194736, 2015.
3. Thomas J. Mowbray, "Cybersecurity: Managing Systems, Conducting Testing, and Investigating Intrusions", Copyright © 2014 by John Wiley & Sons, Inc, ISBN: 978 - 1-118 -84965 -1.
4. James Graham, Ryan Olson, Rick Howard, "Cyber Security Essentials", CRC Press, 15-Dec 2010.
5. Anti- Hacker Tool Kit (Indian Edition) by Mike Shema, McGraw-Hill Publication.

Semester - I
Course Code: BCA/102/MN-1D
Course Title: Principles of Management (4 Credit)

Course Objectives:

To explain the evolution of Management and its principles, discuss the functions of management and their importance in business and propose the application of the principles of management in an organizational setup

Learning Outcomes:

After the completion of this course, the students will be able to:

- Learners will absorb various management concepts such as planning, organizing, implementing, staffing, coordinating, controlling, motivating and Managerial Grid
- Learners will recognize the human skills and conceptual skills as per industry requirements about basic management skills.
- Learners will diagnose various styles and qualities of efficient leadership, Coordination, Controlling, Green Management and Corporate Social Responsibility.

Theory: 60 Lectures

Unit-I: Introduction to Management: Definition, Features of Management, Management Functions, Management as a Process, Importance of Management, Management and Administration, Functional Areas of Management, Managerial Skills, Roles of a Manager, Levels of Management, Management as a Science as well as an Art.

Unit-II: Planning: Meaning and Definition, Features, Steps in Planning Process, Approaches, Principles, Importance, Advantages and Disadvantages of Planning, Types of Plans, Types of Planning, Management by Objectives.

Unit-III: Decision-Making: Meaning, Characteristics, Decision-Making Process, Guidelines for Making Effective Decision, Types of Decisions.

Unit-IV: Organization and Organization Structure: Organizing Process – Meaning and Definition, Characteristics, Process, Need and Importance, Principles, Span of Management. Organization Chart – Types, Contents, Uses, Limitations, Factors Affecting Organizational Chart, Organizational Structure – Line Organization, Line and Staff, Functional, Project, Matrix and Virtual.

Unit-V: Staffing: Meaning, Nature, Importance, Staffing Process – Manpower Planning, Recruitment, Selection, Orientation and Placement, Training, Remuneration, Performance Appraisal, Promotion and Transfer.

Unit-VI: Direction and Supervision: Direction – Definition, Nature, Need and Importance, Principles of Directing. Supervision – Role and Functions of a Supervisor, Effective Supervision, Direction and Supervision.

Unit-VII: Controlling: Meaning, Features, Importance, Control Process, Characteristics of an Effective Control System, Types of Control.

Unit-VIII: Co-ordination: Definition, Characteristics, Essentials, Types and Techniques, Principles, Obstacles and Needs.

Unit-IX: Motivation and Leadership: Motivation: Concept, Theories – Classical and Modern, Importance, Financial and Non-Financial Motivation, Positive and Negative Motivation, Group Motivation. Leadership: Definition, Meaning, Factors, Theories, Principles and Leadership Styles.

Unit-X: Communication: Definition, Meaning, Nature, Communication Process, Types and Barriers to Communication.

Text Books/Reference Books/Online Resources:

1. Management Today Principles& Practice- Gene Burton, Manab Thakur, Tata McGraw-Hill,Publishing Co. Ltd.
2. Management – James A. F. Stoner, Prentice Hall, Inc .U.S.A.
3. Management: Global Prospective –Heinz Weihrich& Harold Koontz, Tata McGrawHill, Publishing Co.Ltd.
4. Essential of Database Management Systems -AlexisLeon, MathewsLeon
5. Management –Task Resp, Practices – PetaDruche “willian Heinemann LTD.
6. Gilbert: Principles of Management, McGraw Hill.

Semester - I
Course Code: BCA/103/MDC-1
Course Title: Mathematics-I (4 Credit)

Course Objectives:

Be able to perform basic computations in higher mathematics. Be able to read and understand middle-level proofs. Be able to write and understand basic proofs. Develop and maintain problem-solving skills.

Learning Outcomes:

After the completion of this course, the students will be able to:

- Understand the foundations of mathematics
- Be able to perform basic computations in higher mathematics

- Be able to read and understand middle-level proofs
- Be able to write and understand basic proofs
- Develop and maintain problem-solving skills
- Use mathematical ideas to model real-world problems
- Be able to communicate mathematical ideas with others
- Have experience using technology to address mathematical ideas

Theory: 60 Lectures

Unit-I: Algebra: Sets, Union and Intersection, Complement, Mapping, Composition, notion of a Group, Ring, Field with simple examples.

Unit-II: Complex Number: Modulus and amplitude, De Moiver's theorem

Unit-III: Polynomials, Division algorithm, Fundamental theorem of classical algebra (Proof not required), Descartes rule of sign and their application, Relation between roots and coefficients; symmetric function of roots, Transformation of polynomial equation, Cardon's solution of cubic equation, Determinants, Addition andMultiplication of Matrices, Inverse of a Matrix ; Solution of linear equations in three variables by Cramer's rule and solution of three line linear equations by matrix inversion methods.

Unit-IV: Vector spaces, Subspaces, Bases and Dimensions, Co-ordinates, Linear Transformation, The Algebra of Linear Transformations.

Unit-V: Vector Algebra: Scalars & vectors, vector addition, linear combination of vectors, condition of colinearity of three points, scalar and vector products, scalar triple product and vector triple product.

Unit-VI: Analytical Geometry: Translation and rotation of rectangular axes, invariants, general equation of second degree-reduction to standard forms and classification. Plane polar equation of a straight line, circle, ellipse, parabola and hyperbola.

Text Books/Reference Books/Online Resources:

1. A Text book of Algebra- B.K. Lahiri & K. C. Roy
2. Linear Algebra- Das & Roy
3. Co-ordinate Geometry- S. L. Loney
4. Differential Calculus- Das and Mukherjee
5. Integral Calculus - Das and Mukherjee

Semester - I
Course Code: BCA/104/SEC-1
Course Title: PC Software (4 Credit)

Course Objectives:
The main objective of this course is to introduce the fundamentals of PC Software and reinforce computer vocabulary, particularly with respect to personal use of computer software, the Internet, networking and mobile computing. It focuses on such computer literacy that prepares students for life-long learning of computer concepts and skills. Students discover why computers are essential components in education, business and society in this course.

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Learning Outcomes: After successfully completing this course, a student will be able to:

- Student will be able to compose, format and edit a word document
- Student will be able to create worksheet, charts in excel
- Student will be able to create presentation in PowerPoint
- Student will be able to send email messages (with or without attachments)
- Student will be able to navigate and search through the internet

Unit-I: Overview of OS: Introduction to DOS, Internal and external commands, batch files (autoexec.bat, config.sys), Line editors, History of Windows (Windows 3.x to up-to-date version), Desktop, user interface action, icon ondesktop, closing windows, renaming icons, resizing windows (maximizing and minimizing). Control panel.

Unit-II: Overview of MS-Word: Introduction to word, Overview, creating, saving, opening, importing, exporting, and inserting files, formatting pages, paragraphs and sections, indents and outdents, creating lists and numbering. Headings, styles, fonts and font size, editing, positioning, viewing texts, searching and replacing text, inserting page breaks, page numbers, bookmarks, symbols, and dates. Using tabs and tablets, header, footer, and printing, mail merge and labels.

Unit-III: Overview of MS-Excel: Getting started with Excel, Worksheet overview, entering information, Editing cell, commands and functions, moving and copying, Inserting and deleting rows and columns, printing worksheet, Creating Charts, Naming range and using statistical, math, and financial function, database in a worksheet, Additionalformatting commands and drawing toolbars, Other commands and functions

Unit-IV: Overview of Power points: Slide creation with PowerPoint, presenting shows for corporate and commercial using power point.

Unit-V: Overview of Cloud Management with Security of Data: Introduction to MS-Teams and MS-SharePoint, Basic Document Management and Document Library, Excel Power Query, MS-Teams App Store, Creating webpages and web parts, Creating forms and Surveys, Restricted Access Management for data security, Integration of MS- SharePoint with MS-Teams.

Unit-VI: Introduction to Internet: Basic internet terms, getting connecting to internet, Internet applications, E-mails, Searching the web, Search engine, and computer viruses.

Text Books/Reference Books/Online Resources:

1. Introduction to Computers with MS-Office-Leon, TMH

Practical:

1. Create a bio data and manipulate text and format using MS-Word.
2. Create a document and design a department invitation using formatting option.
3. Create a document and to insert picture in right side and related information in left side using page layout option in MS-Word.
4. Create text manipulation and mathematics equation with scientific notations.
5. Create your college class timetable using table option in MS- Word.
6. Create a Student mark analysis table using table option in MS- Word.
7. Create a Student mark analysis table and apply formula to Total, Average using formula function in MS-Word. And then convert the Student Mark Table into text using convert option in MS- Word.

8. To write a Student mark statement letter and merge the draft with student's academic database using mail merge on a letter head in MS-Word.
9. To draw a flowchart to find sum of two numbers using drawing toolbars in MS-Word.
10. To create a flowchart for course chart using drawing toolbars in MS-Word.
11. Do the following task in MS Excel

    a. Create a Blank workbook
    b. Save the Workbook as ExcelLabOne.xlsx in Document > MSExcelLabs folder
    c. Type "Working with Excel Math" in cell A1
    d. Merge and Center the text in cell A1 to the range A1:E1
    e. Apply Title style to the range A1:E1
    f. Type Addition in cell A2
    g. Type Subtraction in cell A3
    h. Type Multiplication in cell A4
    i. Type Division in cell A5
    j. Auto-fit the contents in column A
    k. Insert a row above row #2
    l. Type "Basic Math" in cell A2
    m. Merge and Center the text in cell A2 to the range A2:E2
    n. Apply Heading 4 to the range A2:E2
    o. Apply the Calculation Style to the range A3:A6
    p. Type = 4 + 6 / 2 in cell B3 then type = (4 + 6) / 2 in cell D3
    q. Type = 6 – 6 * 2 in cell B4 then type = (6 – 6) * 2 in cell D4
    r. Type = 2 * 2 + 6 in cell B5 then type = (2 * 2) + 6 in cell D5
    s. Type = 9 / 3 + 4 in cell B6 then type = (9 / 3) + 4 in cell D6
    t. Press Ctrl + ~ then press Ctrl + ~ again
    u. Save and submit ExcelLabOne.xlsx to your instructor

12. Use of Formulas Sum, Average, If, Count, Counta, Countif & Sumif

Roll No	Student Name	Bengali	English	Math	Physics	Chemistry	Total	Average	Grade
1	RAM	20	10	14	18	15	77	15.4	A
2	ASHOK	21	12	14	12	18	?	?	?
3	MANOJ	33	15	7	14	17	?	?	?
4	RAJESH	15	14	8	16	20	?	?	?
5	RANJANA	14	17	10	13	18	?	?	?
6	POOJA	16	8	20	17	15	?	?	?
7	MAHESH	18	19	3	10	14	?	?	?
8	ASHUTOSH	19	20	7	14	18	?	?	?
9	ANIL	22	13	8	12	19	?	?	?
10	PREM	26	12	10	11	27	?	?	?

Q.a Find the Total Number & Average in all Subjects in Each Student.
Q.b Find Grade Using If Function - If Average Greater >15 then "A" Grade otherwise "B" Grade

Q.c How Many Student "A" and "B" Grade Use of Countif

Q.d Student Ashok and Manoj Total Number and Average Use of Sumif

Q.e Count how many Students Use of Counta

Q.f How Many Student Bengali & English Subject Number Grater Then > 20 and <15 Use of Countif

13. Use of Formulas - Product, If, Counta, Countif, Sumif

SRNO	ITEMS	QTY	RATE	AMOUNT	GRADE
1	AC	20	40000	800000	Expensive
2	FRIDGE	30	20000	?	
3	COOLER	15	10000	?	
4	WASHING MACHINE	14	15000	?	
5	TV	18	20000	?	
6	FAN	17	2000	?	
7	COMPUTER	10	25000	?	
8	KEYBOARD	5	250	?	
9	MOUSE	25	100	?	
10	PRINTER	30	12000	?	

Q. a Using of Product Fomula for Calculate Amount = Qty*Rate

Q. b How Many Items in a List

Q. c How Many Items qty Greate Then > 20 and Less Then <20

Q. d Calculate Item Computer Qty, Rate and Amount using Sumif Formula

Q. e If Items Amount is Greater > 500000, Then Items "Expensive" otherwise "Lets Buy it".

14. Sales Report (Use of Formulas - Sum, If, Counta, Countif, Sumif, Vlookup, Lookup)

SALESMAN	JAN	FEB	MAR	APR	MAY	JUNE	SALES	TARGET	RESULT
RAMESH	2000	1500	300	1400	1000	1400	7600	10000	NOT ACHIVED
RAKESH	5000	1200	500	1200	1200	2800	?	12000	?
RAHUL	3000	800	1200	3000	1500	3500	?	18000	?
POOJA	1000	900	1800	5000	1400	1200	?	10000	?
MANOJ	500	1000	2300	8000	1700	1400	?	12000	?
ASHOK	800	500	2400	1900	1800	1800	?	10000	?
AJEET	1200	1400	1500	700	2500	7000	?	12000	?
ALOK	1500	1800	1800	1800	300	1500	?	10000	?
AMRIT	1800	2500	1700	1500	2800	1800	?	12000	?
SURENDRA	200	3000	1900	1200	1500	3000	?	10000	?
SHASHI	1600	1200	2000	800	1700	800	?	10000	?

15. Create an excel worksheet of your own and insert data. Then create a Pie chart for the data.
16. Create an excel worksheet of your own and insert data. Then create a line chart for the data.
17. Create an excel worksheet of your own and insert data of students with their percentage of marks. Now implement conditional formatting to distinguish students percentage like the following.

(percentage>=60 Green. percentage>=45 Yellow, percentage<34 RED)

18. Create an excel worksheet of your own and insert student data in it. Now implement freeze pane to lock column heading, and student roll and name.
19. Do the following task in power point.
    a. Open a Blank presentation
    b. Save the presentation as PowerPointLabOne.pptx
    c. Add a Title to the first slide: the name of your college
    d. Type your first name and last name in the Subtitle section
    e. Add a New Slide which has a Title and Content
    f. Add a title to the second slide “My Future Goals”
    g. In the Content section of the second slide, add at least three Personal Goals
    h. Right click on the second slide from the left panel, then choose Duplicate Slide
    i. Highlight the text in the Content area of the third slide. Under the Home tab, click Convert to SmartArt, then choose Basic Cycle
    j. Change the SmartArt Colors to Colorful—Accent Colors
    k. Change the SmartArt Styles to 3D Polished
    l. From the left panel, drag the third slide between the first and second slide
    m. Change the layout of the third slide, the slide that does not have the SmartArt, to Comparison
    n. Leave the title “My Future Goals”
    o. In the head of the first column, type “Goals in College,” then center the heading
    p. In the head of the second column, type “Goals after College,” then center the heading
    q. Add at least three goals in each section
    r. Make sure that slide #3 is selected from the left panel, then add a New Slide
    s. Change the layout of the new slide to Blank
    t. Insert a Graduation Online Picture from the Office ClipArt—Choose any image of your choice
    u. Change the ClipArt size to 3” X 3” and position it in the middle of the slide
    v. Apply the Wisp Design Theme
    w. Save and upload PowerPointLabOne.pptx to your instructor
20. Create a five (5) slide presentation about your City.
21. Create a five (5) slide presentation for Environment awareness.
22. Create a five (5) slide presentation about tourist spots in Bankura.
23. Create a five (5) slide presentation about Prospects of BBA (H) course.
24. Create a five (5) slide presentation on prevention of Road accident.
25. Create a five (5) slide presentation on prevention of pollution in your area.
26. MS-Team
    a. Create a new Team.
    b. Add 2 owners.
    c. Add 3 Members.
    d. Restrict access of members to only view file with visitor category.
    e. Provide edit access to owners.
    f. Create 2 folders named Confidential and All documents.

g. Restrict access of confidential folder to only the owners, the members should not be able to view the contents of the folder name Confidential.
h. Owners should be able to view all folders.

27. Power Query
    a.
        i. Connect to Employee Data file – (Multiple Raw Files to be made available in Lab)
        ii. Replace missing gender & department values
        iii. Remove employees without salary
        iv. Extract employee's country and remove address column
        v. Extract year of join
        vi. Publish data to Excel
    b.
        i. Connect to web source, the database of Member of Parliament of India
        ii. Make a power query to automatically update the database, whenever any alteration in the list.
        iii. Make a graph show case the various metrics in the list, eg, who has the max MPs, who has the max qualifies MPs, who has the most rich MPs etc
    c.
        i. Connect to the Share market live database and make a connection auto refresh, every 30 seconds
        ii. The report should show top 10 stocks and bottom 10 stocks

28. Website using SharePoint and MS – Teams
    a. Create a Website of your college and include the weather of the 4 metro cities with auto update
    b. Provide only view access to 3 people
    c. Include point 2 from the power query as graphs in your website.
    d. Include point 3 from the power query as excel input to display the result.

Semester - II
Course Code: BCA/201/MJC-2
Course Title: Programming and Problem Solving through C-Language (4 Credit)

Course Objectives:
The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also by learning the basic programming constructs they can easily switch over to any other language in future.

Learning Outcomes:
- After the completion of this course, the students will be able to develop applications through C Programming Language.

Theory: 60 Lectures

Unit-I: Introduction to C: History of C, Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C.

Unit-II: Data Types, Variables, Constants, Operators and Basic I/O: Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic,

Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf()), Using Basic Header Files (stdio.h, conio.hetc).

Unit-III: Expressions, Conditional Statements and Iterative Statements: Simple Expressions in C (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Unit-IV: Functions and Arrays: Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.

Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two- dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays

Unit-V: Derived Data Types (Structures and Unions): Understanding utility of structures and unions, declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

Unit-VI: Pointers in C: Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring and initializing references, Using references as function arguments and function return values.

Unit-VII: Memory Allocation in C: Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, storage of variables in static and dynamic memory allocation.

Unit-VIII: File I/O, Preprocessor Directives: Opening and closing a file, Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives.

Text Books/Reference Books/Online Resources:

1. C: The Complete Reference, Herbtz Schildt, Fourth Edition, McGraw Hill.2003
2. Programming in C - B. S. Gottfried (Sahaum Series)
3. Programming in ANSI C- E. Balaguruswami (TMH)
4. Let us C, Y Kanetkar, BPB

Practical:

1. Develop a program to solve simple computational problems using arithmetic expressions and the use of each operator leading to the simulation of a commercial calculator. (No built-in math function)?
2. Write a C program to find all odd numbers between 150 to 300.
3. Write a C program to find LCM of two (2) number.
4. Write a C program to find all 4-digit number which are divisible by 5, 7 and 3.

5. Develop a program to compute the roots of a quadratic equation by accepting the coefficients. Print appropriate messages?
6. Write a C program to print the following pattern.

```
*
* *
 * * *
* * * *
 * * * * *
```

7. Write programs to display each of the following patterns.

```
* * * * *
* * * *
* * *
* *
*
```

8. Write programs to display each of the following patterns.

```
* * * * * * *
* * *   * * *
* *       * *
*           *
```

9. Write programs to display each of the following patterns.

```
1
2 6
3 7  10
4 8  11  13
5 9  12  14  15
```

10. Design and develop an algorithm to find the reverse of an integer number NUM and check whether it is PALINDROME or NOT. Implement a C program for the developed algorithm that takes an integer number as input and output the reverse of the same with suitable messages. Ex: Num: 2014, Reverse: 4102, Not a Palindrome
11. Design and develop a C program to read a year as an input and find whether it is leap year or not. Also consider end of the centuries.
12. Draw the flowchart and Write a C Program to compute Sin(x) using Taylor series approximation given by Sin(x) = x - (x3/3!) + (x5/5!) - (x7/7!) + …….
13. Design develop and Write c program compute cosine (x) .(hint: 1 - (1/2!)x2 + (1/4!)x4 - (1/6!)x6) using Taylor series.
14. Write a program in C to find the factorial of a number without using recursive C function.
15. Write a c –code to determine whether the given number is prime or not using function.
16. Develop a program to find the reverse of a string and check for palindrome or not Display appropriate messages?
17. An Electricity board charges the following rates for the use of electricity: for the first 200 units 80 paise per unit: for the next 100 units 90 paise per unit: beyond 300 units Rs. 1 per unit. All users are changed a minimum of Rs. 100 as meter charge. If the total amount is more than Rs. 400, then an additional surcharge

of 15% of total amount is charged. Write a program to read the name of the user, number of units consumed and print out the charges.

18. Implement using functions to check whether the given number is prime and display appropriate messages (No built-in math function)
19. Develop a program to introduce 2D Array manipulation and implement Matrix multiplication and ensure the rules of multiplication are checked.
20. Write a C program to implement 2D array using pointer.
21. Develop a program to compute Sin(x) using Taylor series approximation. Compare your result with the built-in Library function. Print both the result with appropriate messages
22. Develop a program to sort the given set of N numbers using any sorting algorithm.
23. Develop a program to find the square root of a given number N and execute for all possible Inputs with appropriate messages. Note: Don't use library function sqrt(n).
24. Implement structures to read, write, and compute average marks and the students scoring above and below the average marks for a class of N students
25. Develop a program using pointers to compute the sum and standard deviation of all elements stored in an array of n real numbers.
26. Write a C program to sort an array of structure (roll, name, total marks) in ascending order of total marks.
27. Implement Recursive functions for binary to Decimal Conversion
28. Write a function to implement string operations such as compare, concatenate, string length. Convince the parameters passing techniques
29. Write a C program to insert five (5) records in a file and display it.
30. Write a C program to insert five (5) records in a file and display it, then find a match of a record provided by the user.

Semester - II
Course Code: BCA/202/MN-2A
Course Title: Introduction to Machine Learning (4 Credit)

Course Objectives:

- To understand the basic theory underlying machine learning.
- To be able to formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To be able to apply machine learning algorithms to solve problems of moderate complexity.
- To apply the algorithms to a real-world problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Appreciate the importance of visualization in the data analytics solution
- Apply structured thinking to unstructured problems
- Understand a very broad collection of machine learning algorithms and problems
- Learn algorithmic topics of machine learning and mathematically deep enough to introduce the required theory
- Develop an appreciation for what is involved in learning from data.

Theory: 60 Lectures

Unit-I: Introduction: Machine Learning Foundations – Overview – Design of a Learning System – Types of Machine Learning – Supervised Learning and Unsupervised Learning – Mathematical Foundations of Machine Learning – Applications of Machine Learning.

Unit-II: Supervised Learning – I: Simple Linear Regression – Multiple Linear Regression – Polynomial Regression – Ridge Regression – Lasso Regression – Evaluating Regression Models – Model Selection – Bagging – Ensemble Methods.

Unit-III: Supervised Learning – II: Classification – Logistic Regression – Decision Tree Regression and Classification – Random Forest Regression and Classification – Support Vector Machine Regression and Classification - Evaluating Classification Models.

Unit-IV: Unsupervised Learning: Clustering – K-Means Clustering – Density-Based Clustering – Dimensionality Reduction – Collaborative Filtering.

Unit-V: Association Rule Learning and Reinforcement Learning: Association Rule Learning – Apriori – Eclat – Reinforcement Learning – Upper Confidence Bound – Thompson Sampling – Q-Learning.

Unit-VI: Neural network: Perceptron, multilayer network, backpropagation, introduction to deep neural network.

Text Books/Reference Books/Online Resources:

1. Machine Learning. Tom Mitchell. First Edition, McGraw- Hill, 1997.
2. Introduction to Machine Learning Edition 2, by Ethem Alpaydin
3. Kevin P. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012.
4. Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Third Edition, 2014.
5. Tom Mitchell, "Machine Learning", McGraw-Hill, 1997.
6. Sebastian Raschka, Vahid Mirjilili,”Python Machine Learning and deep learning”, 2nd edition, kindle book, 2018.
7. Carol Quadros,”Machine Learning with python, scikit-learn and Tensorflow”, Packet Publishing, 2018.
8. Gavin Hackeling,” Machine Learning with scikit-learn”, Packet publishing, O’Reily, 2018.
9. Stanford Lectures of Prof. Andrew Ng on Machine Learning

Semester - II
Course Code: BCA/202/MN-2B
Course Title: Data Security and Privacy (4 Credit)

Course Objectives:
The course will cover major developments in statistical data privacy and security, fundamentals of state-of-the-art privacy methods used in the digital world, as well as recent research trends in privacy and security. The course is intended to: (i) provide students with breadth in the areas of data privacy and security through lectures andhomework assignments; and (ii) enable students to specialize in a privacy-related or security-related topic.

Learning Outcomes:
After the completion of this course, the students will be able to:

- To know the concept of Data Security and privacy including personally private information, potential violations of privacy due to security mechanisms, and describe how privacy protection mechanisms run in conflict with security mechanisms.
- To know how an attacker can infer a secret by interacting with a database.
- To know how to set password refresh policy.
- Describe the consequences of data retention policies.

Theory: 60 Lectures

Unit-I: Fundamentals of Data Security & Privacy: Definition, difference between security and privacy, Databases and Exploratory Data Analysis, Data Representation and Storage, Authentication and Authorization, Database Security.

Unit-II: Threats: Definition, types of malware, how malware spread, malware protection, password protection, E-mail security.

Unit-III: Anonymization: Linkage and re-identification attacks, k-anonymity, l-diversity, t-closeness, Implementing anonymization, Anonymizing complex data, Privacy and anonymity in mobile environments.

Unit-IV: Differential Privacy (DP): Formalism and interpretation of DP, Fundamental DP mechanisms and properties, Interactive and non-interactive DP, DP for complex data, Local Differential Privacy (LDP).

Unit-V: Security and Privacy in AI and Machine Learning (AI/ML): Machine Learning (ML) background, Adversary modeling in AI/ML, Poisoning, evasion, and backdoor attacks, Test-time attacks: Model inversion, model stealing, **membership** inference, adversarial examples. Architectures and algorithms for privacy-preserving machine learning

Text Books/Reference Books/Online Resources:

1. Data Privacy and Security (Hb) by Salomon D., Springer
2. Data Protection and Privacy Implementation: India Perspective, Rk Dubey, Ajay Kr Verma

Semester - II
Course Code: BCA/202/MN-2C
Course Title: Cyber Law in India (4 Credit)

Course Objectives:
The objective of the courses to

- Concepts of Technology and Law.
- Providing elementary understanding the authorities under IT Act.
- Penalties & Offences under IT Act.
- Cyber Space Jurisdiction.
- Scope of Cyber Laws.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Technicalities of law in Cyber World.
- Extensive knowledge regarding jurisdictional issues in IT Act.
- Various important national and international cyber laws.
- Understands the scope of Cyber Law.

- The students are able to understand the basic concept of International Technology.

Theory: 60 Lectures

Unit-I: Introduction: History of Internet and World Wide Web, Need for cyber law, Cyber crime on the rise, Important terms related to cyber law.

Unit-II: Cyber law in India: Need for cyber law in India, History of cyber law in India, Information Technology Act, 2000, Overview of other laws amended by the IT Act, 2000, National Policy on Information Technology 2012.

Unit-III: Information Technology Act: Overview of the Information Technology Act, 2000

Unit-IV: Rules issued under the IT Act: Overview of Rules issued under the IT Act, 2000

Unit: V: Electronic commerce and Electronic contracts.

Unit: VI: Cyber crimes and Cyber Frauds: Definition of cyber crime, First Cyber crime, Types of cyber frauds, Cyber frauds in India, Preventive measures, Cyber crimes, Who commits cyber crimes?, Penalties and offences under the IT Act, 2000, Offences under other legislations, Investigation of cyber crimes in India.

Unit-VII: Regulatory Authorities: Department of Electronics and Information Technology, Controller of Certifying Authorities (CCA), Cyber Appellate Tribunal, Indian Computer Emergency Response Team (ICERT),

Unit-VIII: Case Laws: State of Tamil Nadu Vs Suhas Katti, Syed Asifuddin and Ors. V. The State of AP. & Anr., P.R. Transport Agency Vs. Union of India (UOI), SMC Pneumatics (India) Private Limited v. Jogesh Kwatra, Ritu Kohli case, Avnish Bajaj Vs. State (N.C.T.) of Delhi.

Text Books/Reference Books/Online Resources:

1. Justice Yatindra Singh: Cyber Laws, Universal Law Publishing Co., New Delhi.
2. Farouq Ahmed, Cyber Law in India, New Era publications, New Delhi.
3. S.R.Myneni: Information Technology Law (Cyber Laws), Asia Law House, Hyderabad.
4. Chris Reed, Internet Law-Text and Materials, Cambridge University Press.
5. Pawan Duggal: Cyber Law- the Indian perspective Universal Law Publishing Co., New Delhi

Semester - II
Course Code: BCA/202/MN-2D
Course Title: System Analysis and Design (4 Credit)

Course Objectives:
This course examines the systems analysis and design process from understanding what a system should do through how a system should be implemented.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Define the system development life cycle.
    - Conducts research on existing systems.
    - Develop plans for the new system.

- Make the feasibility study about the system.
  - Explore the technical risks involved in the system's and technical possibilities.
  - Scheduling with using GANTT and PERT techniques.
  - Evaluates the economic self-sufficiency whether to install the system.
- Carry out the system analysis.
  - Identifies problems in the system.
  - Determine the cause of the problem in the system.
  - Find a solution of the problem in the system.

Theory: 60 Lectures

Unit-I: Basic Concept of Systems: Basic Concept of Systems The System: Definition and Concepts; Elements of a System: Input, Output Processor, Control, Feedback, Environment, Boundaries and Interface; Characteristics of a System; Types of systems -Physical and Abstract System, Open and Closed Systems, Man-made Systems; Information and its categories

Unit-II: Information System and System Analyst: Information Systems: TPS, OAS, MIS, DSS, ESS; System Analyst: Role and need of system analyst, System Analyst as an agent of change.

Unit-III: System Development Life Cycle: Introduction to SDLC, Various phases: study, analysis, design, development, testing, implementation, maintenance; System documentation: Types of documentation and their importance.

Unit-IV: System Planning and Information Gathering: Initial Investigations, Identification of user needs, Project Identification and Selection; Needs of Information Gathering, Determination of requirements, Information gathering tools: interviews, group communication, questionnaires, presentations and site visits.

Unit-V: Feasibility Study: Definition, Importance of feasibility study, Types of feasibility study, System selection plan and proposal, Prototyping, Cost-Benefit Analysis: Tools and Techniques.

Unit-VI: Tools for System Analysis: Data Flow Diagram (DFD), Logical and Physical DFDs, Developing DFD; System Flowcharts and Structured charts, Structured English, Decision trees and Decision tables.

Unit-VII: System Design: Module specifications, Module Coupling and cohesion, Top-down and bottom-up design; Logical and Physical design, Structured design.

Unit-VIII: Input and Output: Input Design: Input data, Input media and devices; Output design; Form Design: Classification of forms, Requirements of Form design.

Unit-IX: System Implementation and Maintenance: Need of System Testing, Types of System Testing, Quality Assurance; System Conversion, Conversion methods, procedures and controls, System evaluation and performance, Maintenance activities and issues.

Unit-X: System Security and Audit: System Security, Security Threats, Risk Analysis, Control measures, System Audit, Disaster Recovery Planning

Text Books/Reference Books/Online Resources:

1. V. Rajaraman, Analysis and Design of Information System, PHI

2. Elias m. Awad: System Analysis and Design
3. 2. Perry Edwards: System Analysis & design Mc Graw Hill

Semester - II
Course Code: BCA/203/MDC-2
Course Title: Mathematics-II (4 Credit)

Course Objectives:
Be able to perform basic computations in higher mathematics. Be able to read and understand middle-level proofs. Be able to write and understand basic proofs. Develop and maintain problem-solving skills.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the foundations of mathematics
- Be able to perform basic computations in higher mathematics
- Be able to read and understand middle-level proofs
- Be able to write and understand basic proofs
- Develop and maintain problem-solving skills
- Use mathematical ideas to model real-world problems
- Be able to communicate mathematical ideas with others
- Have experience using technology to address mathematical ideas

Theory: 60 Lectures

Unit-I: Differential Calculus: Limit of a function and continuity. Fundamental properties of continuous functions (proofs not required); Derivative and Differential-Geometric meaning, Rules of Differentiation. Successive differentiation.

Unit-II: Rolle's theorem, Mean-Value theorems, Taylor's and Maclaurin's theorems with Cauchy's and Lagrange's forms of remainder; Taylor's series. Functions of several variables. Partial Derivatives. Total Differential. Euler's theorem on homogeneous functions of two variables. Application to plane curves.

Unit-III: Integral Calculus: Rules of Integration of Indefinite Integrals, Solution of Definite Integrals and their elementary properties. Idea of improper integrals.

Unit-IV: Differential Equations: order, degree, solution and formation of a differential equation. Standard techniques of solving a linear differential equation with constant coefficients. Cauchy's and Ligendre's Liner Differential Equations with variable coefficients.

Unit-V: Sequence and Series: Bounded and unbounded sequences, Convergence or divergence of a sequence, Behaviour of monotone sequences, Algebra of convergent sequences, Cauchy sequence, Cauchy's general principle of convergence, Infinite series, it's convergence and sum, series with positive terms and standard tests of convergence (without proofs), Alternating Series, Leibniz Test, Absolute convergence, Rearrangement of absolutely convergent series, Test of convergence of Abel and Dirichlet (without proofs)

Text Books/Reference Books/Online Resources:

1. Differential Equations - Sheplay I. (John Wiley & Sons, Inc)

2. Linear Algebra - Kenneth Hoffman & Ray Kunze (PHI)
3. Mathematical Analysis - S. C. Malic (Wiley Eastern Limited)
4. Differential Calculus – Das and Mukherjee
5. Integral Calculus – Das and Mukherjee

Semester - II
Course Code: BCA/204/SEC-2
Course Title: Front End Web Design and Development (4 Credit)

Course Objectives:
To introduce the basic concepts and techniques of client-side web programming and too enable the students to develop simple, interactive, and stylish websites using HTML, CSS and JavaScript.

Learning Outcomes:
After the completion of this course, the students will be able to:

- After studying this course, students will be able to build websites using the elements of HTML.
- After studying this course, students will be able to build interactive and stylish websites using client-side programming techniques with CSS and JavaScript.
- After studying this course, students will be able to learn to validate client-side data.
- After studying this course, students will be able to define the structure and contents of the website using different features of CSS.

Theory: 60 Lectures

Unit-I: Introduction: Introduction to internet and web design. Basic concepts of web architecture.

Unit-II: HTML: Introduction to hypertext mark-up language (html), creating web pages, lists, hyperlinks, tables, web forms, inserting images, frames.

Unit-III: Cascading style sheet (CSS): Concept of CSS, creating style sheet, Importing style sheets, CSS properties, CSS styling (background, text format, controlling fonts), CSS rules, Style Types, CSS Selectors, CSS cascade, working with block elements and objects, working with lists and tables, CSS id and class, box model (introduction, border properties, padding properties, margin properties).

Unit-IV: Basics of JavaScript: Document object model, data types and variables, functions, methods and events, controlling program flow, built-in objects and operators, validations.

Text Books/Reference Books/Online Resources:

1. Nixon, R., Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5, O'Reilly, 2018.
2. Powell, T.A. HTML & CSS: The Complete Reference, 5th edition, Tata McGrawHill, 2017.
3. Duckett, J., JavaScript and JQuery: Interactive Front-End Web Development, Wiley, 2014.
4. Boehm, A., & Ruvalcaba, Z., Murach's HTML5 and CCS, 4th edition, Mike Murach & Associates, 2018.
5. Ivan Bayross, Web Enabled Commercial Application Development Using Html, Dhtml, Javascript, Perl CGI, BPB Publications, 2010

Practical:

1. Create an HTML document with following formatting – Bold, Italics, Underline, Colors, Headings, Title, Font and Font Width, Background, Paragraph, Line Brakes, Horizontal Line, Blinking text as well as marquee text.
2. Create an HTML document with Ordered and Unordered lists, Inserting Images, Internal and External linking
3. Create an HTML document for displaying the current semester's timetable.
4. Create a website with horizontal and vertical frames. Top horizontal frame needs to show your college's name and logo. Bottom horizontal frame is to be split into two vertical frames. The left frame has hyperlinks to pages related to faculty, courses, student activities, etc. The right frame shows the corresponding webpage based on the link clicked on the left frame.
5. Create a student registration form using HTML which has the following controls and make an interactive content presentation using CSS:
   I. Text Box II. Dropdown box III. Option/radio buttons
   IV. Check boxes V. Reset and Submit button
6. Create a webpage for your department with a drop-down navigation menu for faculty, courses, activities, etc.. Implement the webpage using styles, rules, selectors etc. learned in CSS ● Write event-driven programs in JavaScript for the following:
   - Enter a number and on click of a button print its multiplication table.
   - Print the largest of three numbers entered by the user.
   - Find the factorial of a number entered by the user.
   - Enter a list of positive numbers using the prompt terminated by a zero. Find the sum and average of these numbers.
7. Create a student registration form using text, radio button, check box, drop down box, text field and all other required HTML elements. Customize the CSS and JavaScript to input and validate all data. Create functions to perform validation of each element, example:
   - Roll number is a 7-digit numeric value
   - Name should be an alphabetical value (String)
   - Non-empty and valid fields like DOB

Semester - II
Course Code: BCA/207/INT-1
Course Title: Summer Internship (4 Credit)

Course Objectives:
Acquire on job the skills, knowledge, and attitude, which are requisite to constitute a professional identity.

Demonstrate professional values and ethical standards.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Student is able to construct the company profile by compiling the brief history, management structure, products / services offered, key achievements and market performance for his / her organization of internship.
- For his / her organization of internship, the student is able to assess its Strengths, Weaknesses, Opportunities and Threats (SWOT).
- Student is able to determine the challenges and future potential for his / her internship organization in particular and the sector in general.

- Student is able to test the theoretical learning in practical situations by accomplishing the tasks assigned during the internship period.
- Student is able to apply various soft skills such as time management, positive attitude and communication skills during performance of the tasks assigned in internship organization.
- Student is able to analyze the functioning of internship organization and recommend changes for improvement in processes.

Guidelines for Summer Training Report Writing

Viva Voce on Summer Training (BCA-S306T) is an important part of BCA course curriculum. It provides an opportunity to BCA students to write a summer training report on latest trends/technology related to software project. Following are guidelines for summer training report writing and assessment:

General Instructions

- Summer training report should not be less than 15 pages.
- Proper guidelines to be followed for preparation of summer training report.
- Proper dress code is mandatory for presenting and attending summer training PPT presentations.
- Attendance is compulsory for all students.
- If a student is absent for his presentation as per schedule, he/she must have assessed later on with reducedweightage in the presentation assessment.
- Always prepare a draft report first and print it out.
- Read it yourself first and correct any typographical or grammatical errors.
- **One copy** of final summer training report must be submitted as a **spiraled report** to the coordinator.

Main Components of a Report

- **Cover page**
  As per the given format.

- **Abstract**
  Abstract should be a narration of Problem Definition, Methodology used in third person's language not exceeding 200 words.

- **Acknowledgement and declaration**
  The Acknowledgement shall be brief and should not exceed one page. The students' signature shall be made at the bottom end above their name typed in capital.

- **Certificate**
  Attach a photocopy of a certificate issued from where training has been done.

- **Table of contents/Index page**
  The table of contents should list all material following it. The abstract, acknowledgement will be included in table of contents but the page numbers in lower case Roman letters are to be accounted for them. Pages starting with Introduction (Immediate after the Table of contents)

should be consecutively numbered using Arabic Numerals.

- **Main Text with conclusions**
  This includes various sections for Summer Training Report.
- **References**

Typing Instructions for Summer Training Report

- **Specification for Fonts**
  - **Font Face: Times new Roman**
  - **Font Size:** As per following preview
    - **Headings (Size 16 Bold)**
    - **Sub-Heading (Size 14 Bold and Italic)**
    - **Contents (Size 12 Normal)**
  - **Line spacing: 1.5**
  - **Text Alignment: Both left and right justified**
- Page Dimensions: Standard A4 size (297mm x 210mm)
- Margins:
  - **Top margin: 0.75”**
  - **Bottom margin: 0.75”**
  - **Left margin: 1”**
  - **Right margin: 0.75”**
- Footer: **Page number should be bottom centered.**
- Sections should be numbered as for example,**1. Introduction** etc.
- Subsections should be numbered as for example,***3.1 Simulation Tool*** etc
- Paragraphs and sentences should be short.
- Start of a paragraph should not be intended, rather, give one-line space between two paragraphs.
- A sub heading at the bottom of a page must have at least two full lines below it or else it should becarried over to the next page.
- The last word of any page should not be split using a hyphen.
- **References**

- References should be in IEEE format, i.e.,
  - Author name(s), title of paper/book, or if journal, its name, volume, issues, page number,publisher, year of publication
- Book titles must be in capitals
- Reference numbers should be marked liberally inside the text of the report-
  - eg., ……as given in [3]
- References should either be in chronological order or in the order inwhich they appear in thetext.

Semester - III
Course Code : BCA/301/MJC-3
Course Title : Digital Logic (4 Credit)
Course Objectives:

- To acquire the basic knowledge of digital logic levels and application of knowledge
- To understand digital electronics circuits.
- To impart how to design Digital Circuits.

Learning Outcomes:
At the end of the course, a student will be able to:

- Convert different type of codes and number systems which are used in digital communication and computer systems.
- Employ the codes and number systems converting circuits and Compare different types of logic families which are the basic unit of different types of logic gates in the domain of economy, performance and efficiency.
- Be able to design and analysis combinational logic circuits.
- Be able to design and analysis sequential logic circuits.

Theory: 60 Lectures

Unit I: Introduction to number system: Digital system, Binary numbers, Number base conversation, Positional number system, Binary, Octal, Hexadecimal and decimal number system, Representation of signed numbers and signed magnitude, Binary arithmetic: Binary Addition, Binary Subtraction, Binary multiplication, Binary division, Complementary number system: 1's and 2's complement, Binary subtraction using1's complement method, binary subtraction using 2's complement, Various binary codes: BCD, Excess-3, Gray code, BCD addition.

Unit II: Boolean algebra and Logic gates: Introduction to Boolean algebra -Addition and Multiplication in Boolean algebra: Binary logic function, Logic gates and Truth tables; AND logic, OR logic, NOT logic, NAND logic, NOR logic, EX-OR logic, EX-NOR logic, Boolean rules and Laws, De-Morgan's theorem.
Unit III: Gate level minimization: Realization of switching function using logic gates -Canonical forms, Standard forms, Sum of product forms, Product of sum forms, universal gates: NAND and NOR gates as universal gates, Realization of Boolean function using universal gates. The map method: Three variable map, Four variable map, Logic expression simplification with grouping cell, Quine-McClusky method, realization combinational circuit using truth tables.

Unit IV: Analysis and Design of combinational circuit: Introduction -Binary Adders: Half adder, Full adder; Binary Subtractor, Half Subtractor, Full Subtractor; Parallel binary adder, Binary comparator or Magnitude comparator, Decoders, Encoders, Multiplexer and De-multiplexer, Parity generator and Parity checkers, Code convertor.

Unit V: Latches and Flip-Flops: Latches: Active High S-R Latch (NOR gate S-R latch), Active Low S-R latch (NAND gate S-R latch), Gated S-R Flip-Flop, D-flip-flop, Edge triggered flip-flop: Edge Triggered S-R FF, Edge Triggered D-ff, Edge Triggered J-K flip-flop, Race condition, Master slave J-K flip-flop.

Unit VI: Counters and Registers: Introduction to Three bit Asynchronous counter, Four bit asynchronous counter, ripple counter, Three Bit synchronous binary up counter, Three Bit synchronous binary down counter, Four Bit synchronous binary up counter, Four Bit synchronous binary down counter, Serial In and Serial Out register, Universal Shift register.

Unit VII: Memory and Programmable Logic: Introduction: Random Access memory, Read only memory, Programmable logic array, Sequential programmable devices.

Text Books/Reference Books/Online Resources:

1. Digital Design, M. Morris Mano, Pearson education.
2. Digital Circuit and Designs -S. Alivahanan, S. Arivazhagan - Vikash Publications.
3. Modern Digital Electronics: R. P Jain.

Practical

1. Identify various ICs and their specification-
    a. AND Gate
    b. OR Gate
    c. NOT Gate
    d. NAND Gate
    e. NOR Gate
    f. Exclusive –OR Gate
2. To realize why NAND gate is known as the universal gate by implementation of –
    a. NOT using NAND
    b. AND using NAND
    c. OR using NAND
    d. NOR using NAND
    e. XOR using NAND
    f. X-NOR using NAND
3. To realize why NOR gate is known as the universal gate by implementation of –
    a. NOT using NOR
    b. AND using NOR
    c. OR using NOR
    d. NAND using NOR
    e. XOR using NOR
    f. X-NOR using NOR
4. Implement Half-Adder using Basic Gates.
5. Implement Half-Adder using NAND Gates.
6. Implement Full-Adder using Basic Gates.
7. Implement Full-Adder using NAND Gates.
8. Implement Half-Subtractor using Basic Gates.

9. Implement Half-Subtractor using NOR Gates.
10. Implement Full-Subtractor using NOR Gates.
11. Implement of the following Boolean function using logic gates in both SOP and POS form-
    a. SOP: $AB+A^{/}B^{/}$
    b. POS: $(A+B)(B+C)(A+C^{/})$
12. Implement the Function: using Basic Gates.
13. Implement the Function: F = ABC+ DEF using IC 7411.
14. Implement a 4-bit Parallel Adder using IC 7483.
15. Conversion of Binary to Grey Code.
16. Conversion of Grey to Binary Code.
17. Implement a circuit that convert a Decimal digit from 8, 4, 2, 1 to 8, 4, -2, -1 code.
18. Implement a 3-bit Odd Parity genaratator using Basic Gates.
19. Implement a 4-bit Even Parity Checker using Basic Gates.
20. Implement a 2-bit Comparator using Basic Gates.
21. Implement a 4-bit Magnitude Comparator using 7485 ICs.
22. Implement a 3x8 decoder using Basic Gates.
23. Implement a 3x8 decoder using NAND Gates only.
24. Implement 2 × 1 Multiplexer using Basic Gates.
25. Implement 4 × 1 Multiplexer using NAND Gates.
26. Verification of State Table of S-R Flip-Flop using NAND and NOR gates.

Semester - III
Course Code : BCA/302/MJC-4
Course Title : Object Oriented Programming using C++ (4 Credit)
Course Objectives:
The course is designed to provide complete knowledge of OOP (C++) language. Students will be able to develop logics which will help them to create programs, applications in C++. Also by learning the basic programming constructs they can easily switch over to any other language in future.
Learning Outcomes:

- After the completion of this course, the students will be able to develop applications through C++ Programming Language.

Theory: 60 Lectures

Unit-I: Evolution of Programming methodologies: Introduction to OOP and its basic features, Basic components of a C++, Program and program structure, Compiling and Executing C++ Program. Selection control statements in C++.

Unit-II: Data types, Expression and control statements Iteration statements in C++, Introduction to Arrays, Multidimensional Arrays, Strings and String related Library Functions.

Unit-III: Functions: Introduction to functions, Passing Data to Functions, Scope and Visibility of variables in Functions, Structures in C++.

Unit-IV: Creating classes and Abstraction: Declaration of classes and objects, data members, member functions,

this Pointer, Friends, Friend Functions, Friend Classes, Friend Scope, and Static Functions, Objects as function arguments, Arrays of objects, returning objects from function, structures and classes,

Unit-V: Constructors and Destructors: Constructors, Basic constructors, parameterized constructors, constructors with default argument, dynamic initialization of objects, copy constructors, dynamic constructors, destructors, constraints on constructors and destructors.

Unit-VI: Operator Overloading: Overloading unary operators, binary operators and arithmetic operators, multiple overloading, comparison operators, conversion between objects and basic types, conversion between objects of difference classes, constraints on type conversion.

Unit-VII: Derived Classes and Inheritance: Derived classes and base classes, defining a derived class, accessing base class member, Protected access specifier, derived class constructors, overriding the member function, class hierarchies, abstract base class, constructors and member function, public and private, access combinations and usage of access specifiers, classes and structures, Multiple Inheritance

Unit-VIII: Pointers: Pointers to objects, Virtual Functions, Polymorphism, Abstract classes.

Unit-IX: Files and streams in C++: Character and String input and output to files, Command Line Arguments and Printer Output.

Unit-X: Standard input and output operations: C++ iostream hierarchy, Standard Input/output Stream Library, Organization Elements of the iostream Library, Programming using Streams, Basic Stream Concepts.

Unit-XI: File input and output: Reading a File, Managing I/O Streams, Opening a File – Different Methods, Checking for Failure with File Commands, Checking the I/O Status Flags, Dealing with Binary Files, Useful Functions.

Unit-XII: Templates: Function templates and class templates

Unit-XIII: Standard Template library: Containers, iterators and application of container classes.

Unit-XIV: Exception handling: Throwing an exception, catching an exception: The try block, Exception handlers, Termination vs. Resumption, Exception specification, re-throwing an exception, uncaught exceptions, Standard exceptions, Programming with exceptions.

Text Books/Reference Books/Online Resources:

1. Object Oriented Programming through C++ E. Balagurusamy, TMH
2. C++: The Complete Reference, Herbtz Schildt, Fourth Edition, McGraw Hill.2003
3. C++ Primer, Stanley B. Lippman, Josee Lajoie, Barbara E. Moo, Published by Addison- Wesley, 5th Edition, 2012

Practical

1. Write a program in C++ to print the sum of two numbers.
2. Write a program in C++ to check the upper and lower limits of integer.
3. Write a program in C++ to swap two numbers.
4. Write a program in C++ to calculate the volume of a cube.
5. Write a program in C++ to convert temperature in Celsius to Fahrenheit.
6. Write a program in C++ to print a mystery series from 1 to 50.
7. Write a language program in C++ which accepts the user's first and last name and print them in reverse order with a space between them.

8. Write a C++ program to print the prime number in between 1to 50 and add the prime no.
9. Write a C++ program to swap two numbers using pointer
10. Write a C++ program to calculate sum of distance and display the result using friend function.
11. Write a program to count no of occurrence of particular character in a text file.
12. Write a C++ program to add two complex numbers overloading "+" operator.
13. Write a C++ program to display number of objects created using static member.
14. Define a class named 'Bank Account' to represent following members:
    Data members: -
    Account Number
    Name of Depositor
    Account Type
    Balance Amount
    Member functions:
    - Initialize members
    Deposit Amount
    Withdraw Amount
    Display Balance
    Write a C++ program to test the Bank Account class for 10 customers.
15. Program to print positive number entered by the user If the user enters a negative number, it is skipped.
16. Program to show the concept of function overloading to calculate area where same name function differs in number of parameter.
17. Program to illustrate the use of constructor member function to initialize and object during its creation.
18. Program to understand use of copy constructor.
19. Program to understand the use of constructor overloading and destructor.
20. Program to illustrate how to derive a class from a base class.
21. Program to implement multilevel inheritance.
22. Program to implement hierarchical inheritance
23. Program to illustrate the concept of overriding member function.
24. Program to demonstrate the use of protected members.
25. Program to implement runtime polymorphism using virtual function.

Semester - III
Course Code: BCA/303/MN-3A
Course Title: Deep Leaning (4 Credit)
Course Objectives:

- To introduce the idea of artificial neural networks and their architecture
- To introduce techniques used for training artificial neural networks
- To enable design of an artificial neural network for classification
- To enable design and deployment of deep learning models for machine learning problems

Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the mathematics behind functioning of artificial neural networks
- Analyze the given dataset for designing a neural network based solution
- Carry out design and implementation of deep learning models for signal/image processing applications
- Design and deploy simple TensorFlow-based deep learning solutions to classification problems

Theory: 60 Lectures

Unit-I: Introduction: Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thresholding logic, Linear Perceptron, Perceptron Learning Algorithm, Linear separability, Convergence theorem for Perceptron Learning Algorithm.

Unit-II: Feedforward Networks: Multilayer Perceptron, Gradient Descent, Backpropagation, Empirical Risk Minimization, regularization, auto encoders.

Unit-III: Deep Neural Networks: Difficulty of training deep neural networks, Greedy layer wise training.

Unit-IV: Better Training of Neural Networks: Newer optimization methods for neural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point problem in neural networks, Regularization methods (dropout, drop connect, batch normalization).

Unit-V: Recurrent Neural Networks: Back propagation through time, Long Short Term Memory, Gated Recurrent Units, Bidirectional LSTMs, Bidirectional RNNs.

Unit-VI: Convolutional Neural Networks: LeNet, AlexNet.

Unit-VII: Generative models: Restrictive Boltzmann Machines (RBMs), Introduction to MCMC and Gibbs Sampling, gradient computations in RBMs, Deep Boltzmann Machines.

Unit-VIII: Recent trends: Variational Autoencoders, Generative Adversarial Networks, Multi-task Deep Learning, Multi-view Deep Learning

Unit-IX: Applications: Vision, NLP, Speech (just an overview of different applications in 2-3 lectures).

Text Books / Reference Books / Online Resources:

1. Deep Learning with Keras byAntonio Gulli and Sujit Pal
2. Nikhil Buduma, “Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithm”, O’Reilly, 2017.
3. Ian Goodfellow, YoshuaBengio and Aaron Courville, “Deep Learning”, MIT Press, 2016.
4. Nikhil Ketkar, “Deep Learning with Python: A Hands-on Introduction”, Apress, 2017.

Semester - III

Course Code: BCA/303/MN-3B

Course Title: Data Warehousing and Data Mining (4 Credit)

Course Objectives:

- Be familiar with mathematical foundations of data mining tools.
- Understand and implement classical models and algorithms in data warehouses and data mining
- Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
- Master data mining techniques in various applications like social, scientific and environmental context.
- Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

Learning Outcomes:

After the completion of this course, the students will be able to:

- Understand warehousing architectures and tools for systematically organizing large database and use their data to make strategic decisions.
- Understand KDD process for finding interesting pattern from warehouse.
- Remove redundancy and incomplete data from the dataset using data preprocessing methods.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Discover interesting patterns from large amounts of data to analyze for predictions and classification.

- Develop a data mining application for data analysis using various tools

Theory: 60 Lectures

Unit-I: Data Warehousing: Introduction- Definition and description, need for data ware housing, need for strategic information, failures of past decision support systems, OLTP vs DWH-DWH requirements-trends in DWH-Application of DWH.

Unit-II: Data Warehousing Architecture: Reference architecture- Components of reference architecture - Data warehouse building blocks, implementation, physical design process and DWH deployment process. A Multidimensional Data, Model Data Warehouse Architecture.

Unit-III: Data Mining: Data mining tasks-Data mining vs KDD- Issues in data mining, Data Mining metrics, Data mining architecture - Data cleaning- Data transformation- Data reduction - Data mining primitives.

Unit-IV: Association Rule Mining: Introduction - Mining single dimensional Boolean association rules from transactional databases - Mining multi-dimensional association rules.

Unit-V: Classification and Prediction: Classification Techniques - Issues regarding classification and prediction - decision tree - Bayesian classification –Classifier accuracy – Clustering – Clustering Methods - Outlier analysis.

Unit-VI: Applications and Other Data Mining Methods: Distributed and parallel Data Mining Algorithms, Text mining-Web mining.

Text Books / Reference Books / Online Resources:

1. Jiawei Han and Micheline Kamber, " Data Mining Concepts and Techniques", Morgan Kaufmann Publishers, USA, 2006.
2. Berson,"DataWarehousing, Data Mining and OLAP", Tata McGraw Hill Ltd, New Delhi, 2004.
3. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining, , Pearson Education.
4. Arun K Pujari,"Data mining techniques", Oxford University Press, London, 2003.
5. Dunham M H,"Data mining: Introductory and Advanced Topics". Pearson Education, New Delhi, 2003.
6. Mehmed Kantardzic," Data Mining Concepts, Methods and Algorithms", John Wiley and Sons, USA, 2003.
7. Soman K. P., DiwakarShyam, Ajay V., Insight into Data mining: Theory and Practice, PHI 2006.

Semester - III
Course Code: BCA/303/MN-3C
Course Title: Indian IT Act (4 Credit)
Course Objectives:

- The course aims at acquainting the students with the Basic concepts of Technology and Law and also puts those concepts in their practical perspective. It also provides an elementary understanding of the authorities under IT Act as well as penalties and offences under IT Act.
- The course aims at providing extensive knowledge regarding IT Act, 2000 and Cyber Space Jurisdiction to the students so that students do not face any difficulty while handling practical cases in future as an advocate.
- The course aims at providing extensive knowledge regarding Issues of Internet Governance and International Organizations and their Roles to the students so that students do not face any difficulty while handling practical cases in future.

Learning Outcomes:
After the completion of this course, the students will be able to:

- In Depth Knowledge of Information Technology Act and Legal Frame Work Of Right To Privacy, Data Security And Data Protection.

- To develop the conceptual understanding of the cyber dispute and its resolution
- To trained the students to deal with cyber crimes cases
- To explain the jurisdictional issues in cyber space.

Theory: 60 Lectures

Unit-I: Salient features of IT Act – 2000: a) Definitions. b) Electronic record and digital signature authentication. c) Electronic governance. d) Various authorities under IT Act and their powers. e) Penalties. f) Offences. g) Miscellaneous.

Unit-II: Impact on other related Acts (Amendments): a) Amendments to Indian Penal Code. b) Amendments to Evidence Act. c) Amendments to Bankers Book Evidence Act. d) Amendments to Reserve Bank of India Act.

Unit-III: Concept of Cyber – space jurisdiction and other principal of jurisdiction: a) Territorial jurisdiction b) Extra Territorial jurisdiction c) Cyber – space jurisdiction.

Unit-IV: Investigation of Cybercrime (Technical Issue).

Unit-V: Amendments: Indian IT Acts amendments.

Unit-VI: Case Laws: Indian IT Acts Case Laws.

Text Books / Reference Books / Online Resources:

1. Raj, Niharikia, Law & Technology, Universal Law Publishing. Information Technology Act, 2000.

Semester - III
Course Code: BCA/303/MN-3D
Course Title: IT Service Management (4 Credit)
Course Objectives:
Objectives Upon completion of this course, students will be able to: Understand how to plan and create a service value stream to make, deliver and support services. Know how relevant ITIL® practices contribute to creating, delivering, and supporting through SVS and value streams.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Have knowledge and critical understanding of the fundamental principles of ITSM.
- Understand how ITSM can operate in an organisation to improve processes.
- Understand best management practice from a technical and non-technical perspective.

Theory: 60 Lectures

Unit-I: IT Service Management: Introduction, What is service management? What are services? Business Process, Principles of Service management: Specialization and Coordination, The agency principle, Encapsulation, Principles of systems, The service Life Cycle, Functions and processes across the life cycle. **Service Strategy Principles -** Value creation, Service Assets, Service Provider Service Structures, Service Strategy Principles. **Service Strategy -** Define the market, Develop the offerings, Develop Strategic Assets, Prepare for execution. **Challenges, Critical Success factors and risks -** Complexity, Coordination and Control, Preserving value, Effectiveness in measurement, Risks.

Unit-II: Service Design: Fundamentals. **Service Design Principles** - Goals, Balanced Design, Identifying Service requirements, identifying and documenting business requirements and drivers, Design activities, Design aspects,

Subsequent design activities, Design constraints, Service oriented architecture, Business Service Management, Service Design Models. **Service Design Processes -** Service Catalogue Management, Service Level Management, Capacity Management, Availability Management, IT Service Continuity Management, Information Security Management, Supplier Management. **Challenges, Critical Success factors and risks -** Challenges, Risks.

Unit-III: Service Transition: Fundamentals. **Service Transition Principles -** Principles Supporting Service Transition, Policies for Service Transition. **Service Transition Processes** - Transition planning and support, Change Management, Service Asses Configuration Management, Service and Deployment Management, Service Validation and Testing, Evaluation, Knowledge Management. **Challenges, Critical Success factors and risks -** Challenges, Critical Success factors, Risks, Service Transition under difficult Conditions.

Unit-IV: Service Operation: Fundamentals. **Service Operation Principles** - Functions, groups, teams, departments and divisions, Achieving balance in service operations, Providing service, Operation staff involvement in service design and service transition, Operational Health, Communication, Documentation. **Service Operation Processes** -Event Management, Incident Management, Request fulfilment, Problem Management, Access Management, Operational activities of processes covered in other lifecycle **phases**. **Challenges, Critical Success factors and risks** - Challenges, Critical Success factors, Risks

Unit-V: Continual Service Improvement(CSI) Principles: CSI Approach, CSI and organizational change, Ownership, CSI register, External and Internal drivers, Service level management, Knowledge management, The Deming cycle, Service Measurement, IT governance, Frameworks, models, standards and quality Systems, CSI inputs and outputs. **CSI Process -** The seven step improvement process. **CSI Methods and Techniques** - Methods and techniques, Assessments, benchmarking, Service Measurement, Metrics, Return on Investment, Service reporting, CSI and other service management processes. **Organising for CSI -** Organisational development, Functions, roles, Customer Engagement, Responsibility model - RACI, Competence and training. **Technology considerations** - Tools to support CSI activities **Implementing CSI** - Critical Considerations for implementing CSI, The start, Governance, CSI and organisational change, Communication Strategy and Plan

Text Books / Reference Books / Online Resources:

1. Service Management, Fourth Edition, J.A. Fitzsimmons and M.J. Fitzsimmons, McGraw Hill.
2. Principles of Service Marketing and Management, Christopher Lovelock and Lauren Wright, Prentice Hall.
3. IT Service Management – A Concise Study, S. A. Kelkar, PHI

Semester - III
Course Code: BCA/304/MD-3
Course Title: Mathematics-III (4 Credit)
Course Objectives:
Be able to perform basic computations in higher mathematics. Be able to read and understand middle-level proofs. Be able to write and understand basic proofs. Develop and maintain problem-solving skills.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Describe a data set including both categorical and quantitative variables to support or refute a statement,
- Apply laws of probability to concrete problems,
- Perform statistical inference in several circumstances and interpret the results in an applied context,
- Use mathematical tools, including calculus and linear algebra, to study probability and mathematical statistics and in the description and development of statistical procedures.
- Perform different accounting problems in Tally.

Theory: 60 Lectures

Unit-I: Probability and Statistics: Permutation and Combinations, Probability, Classical definition of probability. Conditional probability. Statistical independence of events. Random variable and its expectation and variance, joint dispersion of attributes.

Unit-II: Collection and presentation of data, Frequency distribution, Measures of central tendency, Measures of dispersion, Binomial, Poisson and Normal distribution.

Unit-III: Bivariate Frequency Distributions (scatter Diagram, Correlation coefficient and its properties, regression lines, correlation index and correlation ratio, rank correlation).

Unit-IV: Multiple linear regression, multiple correlation, partial correlation (for 3 variables only).

Unit-V: Random sampling, expectations and standard error of sampling mean. Expectation and standard error of sampling proportions.

Unit-VI: Test of significance based on t, F, and CHI square distribution.

Unit-VII: Numerical Methods and Algorithms Solution of non-linear equations: Bisection, Newton-Raphson, Regular-Falsi and Secant method. Interpolation and approximation- Lagrange Interpolation, Newton's Forward Interpolation and Newton's backward Interpolation methods.

Unit-VIII: Integration: Trapizoidal and Simpson's 1/3 rules.

Unit-IX: Solution of linear equations: Gaussian elimination, Gauss Sheidal method.

Unit-X: Solution of different equations; Eular's, Taylor's series, Runge-kutta (order-2)

Text Books / Reference Books / Online Resources:

1. C Language and Numerical Methods C Xaviers, New Age International
2. Fundamentals of Statistics – Goon, Gupta, DasGupta
3. Statistical Methods (vol 1 and 2) – N.G. Das.
4. Mathematical and Statistics – Ajay Goel and Alka Goel; Taxmann.
5. Statistics – Sancheti and Kapoor; Sultan Chand & Sons.

Semester - III
Course Code: BCA/305/SEC-3
Course Title: Digital Marketing (4 Credit)
Course Objectives:
To understand the basic Concepts of Digital marketing and the road map for successful Digital marketing strategies.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Develop and execute a marketing plan, incorporating all elements of the marketing mix, segmentation and positioning strategies and other elements.
- Understanding of the role of both digital and traditional media in marketing, and the intersection of online and offline strategies and tactics.
- Guide the development of a digital presence from a marketing point of view.
- Proficient in marketing analytics and quantitative evaluation of the marketing environment.
- Working knowledge of website design and development.

Theory: 60 Lectures

Unit-I: Introduction to Digital Marketing: Meaning of Digital Marketing, Differences from Traditional Marketing, Return of Investments on Digital Marketing vs. Traditional Marketing, E Commerce, Tools used for successful marketing, SWOT Analysis of Business for Digital Marketing, Meaning of Blogs, Websites, Portal and Their Differences, Visibility, Visitor Engagement, Conversion Process, Retention, Performance Evaluation.

Unit-II: Search Engine Optimization (SEO): On page Optimization Techniques, Off Page Optimization Techniques, Preparing Reports, Creating Search Campaigns, Creating Display Campaigns.

Unit-III: Social Media Optimization (SMO): Introduction to Social Media Marketing, Advanced Facebook Marketing.

Unit-IV: Social Media Marketing: Word press Blog Creation, Twitter Marketing, LinkedIn Marketing, Instagram Marketing, social media Analytical Tools. Search Engine Marketing: Meaning and Use of Search Engine Marketing, Tools used — Pay Per Click, Google Adwords, Display Advertising Techniques, Report Generation.

Unit-V: Website Traffic Analysis, Affiliate Marketing and Ad Designing: Google Analytics, Online Reputation Management, Email Marketing, Affiliate Marketing, Understanding Ad Words Algorithm, Advertisement Designing.

Text Books / Reference Books / Online Resources:

1. Digital Marketing –Kamat and Kamat-Himalaya
2. Marketing Strategies for Engaging the Digital Generation, D. Ryan,
3. Digital Marketing, V. Ahuja, Oxford University Press
4. Digital Marketing, S.Gupta, McGraw-Hill
5. Quick win Digital Marketing, H. Annmarie, A. Joanna, Paperback edition

Semester - IV

Course Code : BCA/401/MJC-5

Course Title : Operating System (4 Credit)

Course Objectives:

To understand the structure and organization of the file system. To understand what a process is and how processes are synchronized and scheduled. To understand different approaches to memory management. Students should be able to use system calls for managing processes, memory and the file system.

Learning Outcomes:

After the completion of this course, the students will be able to learn

- How Operating System is Important for Computer System.
- To make aware of different types of Operating System and their services.
- To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- To know virtual memory concepts.
- To learn secondary memory management
- Write different shell and awk scripts

Theory: 60 Lectures

Unit-I: Introduction: Definition and functions of Operating System, Evolution of Operating Systems-Simple Batch Operating Systems, Multi-programmed Batched Operating Systems, Time- Sharing operating Systems, Personal Computer Operating Systems, Multi-processor Operating Systems, Distributed Systems, Real-Time Systems; Operating system structures-Layered approach, The kernel based approach, The virtual machine approach.

Unit-II: Operating System Architecture: Operating System as an Extended Machine, Layered Approach, Micro-Kernels, UNIX Kernel Components, Modules, Introduction to Virtual Machines, Virtual Environment & Machine Aggregation, Implementation Techniques.

Unit-III: Process Management: Process, Process State, Process Control Block, Process Scheduling, Operation on processes, Co-operating Processes, Threads.

Unit-IV: CPU Scheduling Algorithms: Basic Concepts of Scheduling: CPU-I/O Burst Cycle. CPU Scheduler, Pre-emptive / non pre-emptive scheduling, Dispatcher, Scheduling Criteria; Scheduling Algorithms, First come First Served Scheduling, Shortest-Job-First Scheduling, Priority Scheduling. Round-Robin Scheduling, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling, Multiple-Processor Scheduling, Real-Time Scheduling; Evaluation of CPU Scheduling Algorithms-Deterministic Modelling, Queuing Models, Simulations, Implementation.

Unit-V: Process Synchronization: Inter process Communication; Basic Structure, Naming: Direct Communication; Indirect Communication, Buffering; The Critical-section problem: Two Process Solution; Multiple Process Solutions; Semaphores; Monitors; Hardware Assistance.

Unit-VI: Deadlocks: System Model, Deadlock Characterization, Deadlock Handling; Deadlock Prevention, Deadlock Avoidance-Safe State, Resource-Allocation Graph Algorithm, Banker's Algorithm; Deadlock Detection-Single Instance of a Resource, Multiple Instances of a Resource and Recovery from Deadlock.

Unit-VII: Memory Management: Logical versus Physical Address Space Swapping; Contiguous Allocation-Single partition Allocation, Multiple Partition Allocation, Fragmentation; Paging-Concept of paging, Page Table Implementation; Segmentation-Concept of Segmentation, Segmentation Hardware, External Fragmentation.

Unit-VIII: Virtual Memory: Need for Virtual Memory Technique; Demand Paging; Page Replacement; Page Replacement Algorithms-FIFO Page Replacement Algorithm, Optimal Algorithm; LRU page Replacement Algorithm; Thrashing-Causes for Thrashing, Working Set Model, Page Fault Frequency.

Unit-IX: File System Interface and Implementation: Concept of a File- Attributes of a File, Operations on Files, Types of Files; Structure of File; File Access Methods-Sequential Access, Direct Access, Indexed Sequential Access; Directory Structure: Single Level Directory, Two Level Directory; Tree Structured Directories; Allocation Methods-Contiguous allocation, Linked allocation, Indexed allocation, Performance comparison; Free Space Management, Directory Implementation.

Unit-X: Input-Output Architecture: I/O Structure, I/O Control Strategies-Program controlled I/O, Interrupt-controlled I/O, Direct memory access; The I/O Address Space.

Unit-XI: Operating Systems in Distributed Processing: Centralized and Distributed Processing, Network Operating System (NOS) Architecture, Functions of NOS, Global Operating System (GOS), Remote Procedure Call (RPC), Distributed File Management.

Unit-XII: Security and Protection: Attacks on Security, Computer Worms, Computer Virus, Security Design Principles, Authentication, Protection Mechanism, Encryption, Security in Distributed Environment.

Unit-XIII: Unix Editors and commands: ed editor, vi editor, Redirections, piping, tees, filters, UNIX utilities: grep, sed, awk, tr etc.

Unit-XIV: Introduction to Shell scripts: Bourne shell, C shell, Shell variables, Scripts, meta-characters and environments,

if and case statements, for, while and until loops.

Unit-XV: Awk programming: Awk arithmetic and variables, Awk built-in variable names and operators, arrays, strings.

Text Books / Reference Books / Online Resources:

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8 th Edition, John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education 1997.
4. W. Stallings, Operating Systems, Internals & Design Principles, 5 th Edition, Prentice Hall of India. 2008.
5. M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.
6. Y. Kanetkar, Unix Shell Programming, BPB.
7. Sumitabha Das, UNIX, concepts and applications, Mcgraw Hill

Practical

1. Write a shell Script to calculate sum of digits of a given number.
2. Write a shell Script to compute LCM of three given number.
3. Write a shell Script to find the first 8 Fibonacci numbers.
4. Write a shell Script to check whether a number is prime or not.
5. Write a shell Script to find out reverse of a four-digit given integer.
6. Write a shell Script to compute the series 1+3+5+7+…. up to 10th term.
7. Write a shell Script to convert Celsius to Fahrenheit & vice-versa.
8. Write a shell Script to compute factorial of an integer.
9. Write a shell Script to compute HCF of two given number.
10. Write a shell Script to find a given no is palindrome or not.
11. Write a shell Script to find the system date is leap year or not.
12. Write a shell Script to find a file (inputted by the user) is present in current directory or not give proper message.
8. Write a shell Script to find who are currently logged in in the system (display only the name).
9. Write a shell Script to merge contents of two files into a file, now find a particular string in the file.
10. Find the difference between echo $PATH; echo "$PATH"; echo '$PATH'; echo *; echo "*"; echo '*';

Semester - IV
Course Code : BCA/402/MJC-6
Course Title : Software Engineering (4 Credit)
Course Objectives:

- To provide the idea of decomposing the given problem into Analysis, Design, Implementation, Testing and Maintenance phases.
- To provide an idea of using various process models in the software industry according to given circumstances.

Learning Outcomes:
At the end of the course students will be able to:

- Basic knowledge and understanding of the analysis and design of complex systems.
- Ability to apply software engineering principles and techniques.

Theory: 60 Lectures

Unit-I: Software Engineering Fundamentals: Definition of software product, Software Engineering Paradigms; Software engineering, Knowledge engineering, and End user development approaches.

Unit-II: System Analysis: An abstraction, Partitioning and projection, Systems specification, Software Requirements Specification (SRS) standards, Formal Specification methods, Specification tools, Flow based, Data based and Object –

Oriented Analysis.

Unit-III: System Documentation: Principles of system documentation, types of documentation and their importance.

Unit-IV: System Planning: Data and fact gathering techniques-Interviewing, communications, presentations and site visit. Feasibility study, feasibility reports, prototyping, cost-benefit analysis-tools and techniques.

Unit-V: Systems Design: Idealized and constrained design, Process oriented design (Game and Sarson and Yourdon notations), Data oriented design (Warnier–Orr, E-R modelling), Object oriented design (Booch approach), Cohesion and Coupling, Design matrices, Design documentation standard.

Unit-VI: Role of CASE Tools: Relevance of CASE Tools, High-end and Low-end CASE Tools.

Unit-VII: Coding and Programming: Choice of programming languages, Mixed language programming and cell semantics, Reengineering legacy systems, Coding standard.

Unit-VIII: Software Quality and testing: Software quality assurance .Types of Software Testing (White Box and Black Box Testing, Unit Testing, Integration Testing, Verification and Validation of Software) , Debugging and Software Reliability analysis , Software quality and matrices, Software maturity model and extensions.

Unit-IX: Software Cost and Time estimation: Functions points, Issues in software cost estimation, Introduction to the Rayleigh curve, Algorithmic cost models (COCOMO, Putnam- Slim, Watson, and Felix), Other approaches to software cost and Size estimation (software complexity, Delphi , costing by analogy).

Unit-X: Software Project Management: Planning software , projects, Work breakdown structures, Integrating software design and project planning ,Software project teams, Projecting monitoring and control.

Text Books / Reference Books / Online Resources:

1. R S Pressman, Software Engineering; A Practitioner's Approach (7th Edition), MGH, 2009
2. P Jalote, An Integrated Approach to Software Engineering (2nd Edition), NPH, 2003
3. R. Mall, Fundamentals of Software Engineering (2nd Edition), PHI, 2004

Semester - IV

Course Code : BCA/403/MJC-7

Course Title : Data Structure & Algorithm through C++ (4 Credit)

Course Objectives:

Able to identify the appropriate data structures and algorithms for solving real world problems. Able to implement various kinds of searching and sorting techniques. Able to implement data structures such as stacks, queues, Search trees, and hash tables to solve various computing problems.

Learning Outcomes:

At the end of the course, the students will be able to:

- Implement basic data structures such as arrays and linked list.
- Programs to demonstrate fundamental algorithmic problems including Tree Traversals, Graph traversals, and shortest paths.
- Implement various searching and sorting algorithms.
- Programs to demonstrate the implementation of various operations on stack and queue.

Theory: 60 Lectures

Unit-I: Analysis of Algorithm: Introduction to Algorithm Design and Data Structures: Design and analysis of algorithm: Algorithm definition, comparison of algorithms. Top down and bottom up approaches to Algorithm design. Analysis of Algorithm (Mathematical Presentation- Frequency count, Complexity measures in terms of time and space). Structured approach to programming.

Unit-II: Elementary Data Structures-I (Arrays): Representation of arrays: single and multidimensional arrays. Address calculation using column and row major ordering. Various operations on Arrays, Vectors. Application of arrays: Matrix multiplication, Sparse polynomial representation and addition, Stacks and Queues: Representation of stacks and queues using arrays, Circular queues, Priority Queue and D-Queue, Applications of stacks and Queues, Conversion from infix to postfix and prefix expressions, Evaluation of postfix - 95 - expression using stacks. Pointers: Definition, Pointer Arithmetic, Array of pointers, Arrays in terms of pointers.

Unit-III: Elementary Data Structures-II (Linked Lists): Linked list: Singly linked list; operations on list, Linked Stacks and Queues: Representation of stacks and queues using Link List. Polynomial representation and manipulation using linked lists. Circular linked lists, doubly linked lists. Generalized list structure. Sparse Matrix representation using generalized list structure.

Unit-IV: Abstract Data types Stacks and Queues: Definition of ADT, Stack ADT (array implementation), FIFO queue ADT (array implementation)

Unit-V: Trees: Binary tree traversal methods: Preorder, In-order, Post-ordered traversal. Recursive Algorithms for above mentioned Traversal methods. Representation of trees and its applications: Binary tree representation of a general tree. Conversion of forest into tree. Threaded binary trees. Binary search tree. : Height balanced (AVL) tree, B-trees.

Unit-VI: Searching, Sorting and Complexity: Selection sort, Insertion sort, Bubble sort, Quick sort, merge sort, Heap sort, Radix sort and their complexity, Searching: Sequential search, Binary Search, Binary Search Tree, ASVL trees, B trees, Searching , sorting and complexity, Searching : Sequential and binary searches, Indexed search, Hashing Schemes. Sorting: Insertion, selection, bubble, Quick, merge, radix, Shell, Heap sort, comparison of time complexity.

Unit-VII: Graphs: Graph representation: Adjacency matrix, Adjacency lists, Traversal schemes: Depth first search, Breadth first search. Spanning tree: Definition, Minimal spanning tree algorithms. Shortest Path algorithms (Prime's and Kruskal 's).

Text Books / Reference Books / Online Resources:

1. Hubbard John. R, "Schaum's outline of Data Structures with C++", Tata McGraw-Hill, 2007.
2. Langsam Y,.Augenstein M. J and Tanenbaum A. M, "Data Structures Using C and C++", Second Edition, Pearson Education, 2007.
3. Kruse R, Tonodo C.L. and Leung B, "Data Structures and Program Design in C", Pearson Education, 2007.
4. Horowitz E, Sahni S and Mehta D, "Fundamentals of Data Structures in C++," Galgotia Publiction, 2009.
5. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
6. Weiss M A, "Data Structures and Algorithm Analysis in C++", Pearson Education, 2007
7. Litvin G, "Programmking with C++ and Data Structures", Vikas Publishing House.

Practical

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.

5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
6. Perform Stack operations using Linked List implementation.
7. Perform Stack operations using Array implementation. Use Templates.
8. Perform Queues operations using Circular Array implementation. Use Templates.
9. Create and perform different operations on Double-ended Queues using Linked List implementation.
10. WAP to scan a polynomial using linked list and add two polynomial.
11. WAP to calculate factorial and to compute the factors of a given no. (i)using recursion, (ii) using iteration.
12. WAP to display fibonacci series (i)using recursion, (ii) using iteration.
13. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion.
14. WAP to create a Binary Search Tree and include following operations in tree: (a) Insertion (Recursive and Iterative Implementation) (b) Deletion by copying (c) Deletion by Merging (d) Search a no. in BST (e) Display its preorder, postorder and inorder traversals Recursively (f) Display its preorder, postorder and inorder traversals Iteratively (g) Display its level-by-level traversals (h) Count the non-leaf nodes and leaf nodes (i) Display height of tree (j) Create a mirror image of tree (k) Check whether two BSTs are equal or not.
15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
16. WAP to reverse the order of the elements in the stack using additional stack.
17. WAP to reverse the order of the elements in the stack using additional Queue.
18. WAP to implement Diagonal Matrix using one-dimensional array.
19. WAP to implement Lower Triangular Matrix using one-dimensional array.
20. WAP to implement Upper Triangular Matrix using one-dimensional array.
21. WAP to implement Symmetric Matrix using one-dimensional array.
22. WAP to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the successor / predecessor of an element, insert an element, inorder traversal
23. Write a function that will concatenate two circularly linked lists, producing one circularly linked list.
24. WAP to implement Breadth First Traversal(BFS)
25. WAP to implement Depth First Traversal(DFS)
26. WAP to implement Prime's Algorithm.
27. WAP to implement Kruskal 's Algorithm.

Semester - IV
Course Code : BCA/404/MJC-8
Course Title : Computer Networks (4 Credit)
Course Objectives:
The course objectives include learning about computer network organization and implementation, obtaining a theoretical understanding of data communication and computer networks, and gaining practical experience in installation, monitoring, and troubleshooting of current LAN systems.
Learning Outcomes:
At the end of the course, the students will be able to:

- Understand and describe the layered protocol model.
- Describe, analyses and evaluate a number of datalink, network, and transport layer protocols.
- Program network communication services for client/server and other application layouts.
- Describe, analyses and evaluate various related technical, administrative and social aspects of specific computer network protocols from standards documents and other primary materials found through research.
- Design, analyses, and evaluate networks and services for homes, data centres, IoT/IoE, LANs and WANs.

Theory: 60 Lectures

Unit-I: Data Communications: Introduction, Communication Systems, Signal and data, Transmission modes, Synchronous and asynchronous transmission, Circuits, channels and multi channeling, Signaling, Encoding and decoding, Error detection

and Recovery, Flow control, Sliding Window, Congestion Management, Multiplexing [FDM, TDM, CDM, WDM] and Spreading [DS. FH], Concept of Modulation, Baseband versus Broadband; Pulse Code Modulation (PCM), Shift Keying [ASK, FSK, PSK, QPSK, DPSK]; Encoding techniques and CODEC; Classification of Modems, Standards and Protocols, Protocols used by Modem to Transfer files, Establishing a Connection (Internet connectivity); Digital Subscriber Loop (DSL)

Unit-II: Communication Network Fundamentals: Introduction, Switching techniques: Circuit Switching, Packet switching, Datagram, Virtual circuit and Permanent Virtual Circuit, Connectionless and connection oriented communication, Message switching, Cell switching (ATM); Telephone network signaling Network topologies, Layering the communication process, Open Systems Interconnection (OSI) model, Data encapsulation; Protocols, services and layering, PDU/SDU; TCP/IP suite, Hour-glass model, Internet Architecture and Protocol overview.

Unit-III: Media Access Control: Introduction, Access Techniques (STDM, FDMA, TDMA, Spread Spectrum techniques and CDMA, DSSS, FHSS), Media Access Control, Aloha and Slotted Aloha, Media Access Control Address, Polling, CSMA, CSMA/CA, CSMA/CD and Reservation Aloha, Digital hierarchies [SONET/SDH]

Unit-IV: Network Components: Introduction, LAN Hardware, LAN Operating Systems, Transmission Media: Guided Media (Twisted pair, Co-axial cable, Optical fiber), Unguided Media (Radio, VHF, microwave, satellite, Infrared); Fiber Optics Communication Components (Source, Channel Detector).

Unit-V: Layered Protocols OSI Model: Goals of layered protocols, network design problems, communication between layers, introduction to standard organizations and the OSI model, standards organizations, Layers of OSI, OSI status.

Unit-VI: Switching and Routing in Networks: Message switching, packet switching, when and when not to use packet switching, packet routing, and packet switching support to circuit switching networks.

Unit-VII: Link Control and MAC Protocols: Framing, Error Detection and Correction; Window-based Flow Control; Logical Link Control, HDLC Protocol, Point-to-Point Protocol (PPP), X.25 CCITT standard for packet data transmission; Media access control, Random Access Techniques, Scheduling Mechanisms.

Unit-VIII: Local Area Network (LAN): LAN topologies and protocols; IEEE 802 Standard; Ethernet (Standard, Fast, Gigabit), Token Ring, FDDI, Wireless LANs (802.11x); Connecting LANs: Repeaters, Bridges, Switches, Routers; Virtual LANs

Unit-IX: Wide Area Network (WAN): Network Layer Addressing and Routing concepts (Forwarding Function, Filtering Function); Routing Methods (Static and dynamic routing, Distributed routing, Hierarchical Routing); Distance Vector Protocol, Link State protocol, Open Shortest Path First (OSPF); Internet Protocol (IP): Addressing & Routing; Internet Control Message Protocol, (ICMP), Address Resolution Protocol (ARP), Dynamic Host Control Protocol (DHCP), Network Address Translation (NAT), IPv6, Mobile IP, Process-to-Process delivery in Transport Layer: User Datagram Protocol (UDP), Transmission Control Protocol (TCP), congestion control, TCP/IP Reference Model and internetworking, example of TCP/IP operations, related protocols, Concept of ports and sockets.

Unit-X: The X.25 and supporting protocols: Features of X.25, Layers of X.25 and the physical layer, X.25 and the data link layer, X.25 standards, X.25 channel options, flow control principles.

Unit-XI: Application Protocols: Client/Server Model, Network File System (NFS), Remote Login: Telnet; File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP); E-mail system: Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP); World Wide Web (WWW), Domain Name System (DNS), DNS servers; Hyper Text system: Hyper Text Transfer Protocol (HTTP), Hyper Text markup Language (HTML)

Unit-XII: Wireless Networks: Radio Communications, Cellular Radio, Mobile Telephony (GSM & CDMA), Satellite Networks (VSAT), Mobile Adhoc Networks (MANET)

Unit-XIII: Security and Management: Cryptography, IPsec, SSL/TLS, PGP, secure HTTP, proxy, firewall, VPN; Simple Network Management Protocol (SNMP), Network policies.

Text Books / Reference Books / Online Resources:

1. Computer Networks- protocols, standards and Interfaces, Black U., P.H.I.
2. Computer Communication Networks, Stallings W., P.H.I.
3. Computer Networks, Tannembaum A. S., P.H.I
4. Data Communication and Computer Networks : Brijendra singh (PHI)
5. Data Communication and Networking, Behrouz A Forouzan, Tata McGraw-Hill, 2008
6. Data Communication and Computer Networks, Rajneesh Agrawal and Bharat Bhushan Tiwari, Vikas Publishing House Ltd., 2005.
7. Introduction to Data Communications and Networking, Tomasi Wayne, Pearson Education, 2007

Practical

1. Open Linux Terminal, explain when we use these networking Command?
   ip, route, arp, whois, ss
2. How to Create two different network using Router in Cisco Packet Tracer
3. How to differentiate between Router & Bridge in Cisco Packet Tracer
4. Socket Programming- UDP Client/Server Socket.
5. How to make Client Server Communication using Socket Progarmming?
6. Open Windows Terminal when we use these following Networking Commands? Ping, tracert, nslookup, ipconfig /all, netstat -a
7. Create a Simple LAN Connection using Cisco Packet Tracer.
8. Set a domain icis.org.in in a DNS Server & Create a WAN Communication from your device (PC, Laptop etc..) using Cisco Packet Tracer.
9. Socket Programming- TCP Client/Server Socket.
10. Write a Program to implement Caesar Cipher.
11. Open Linux Terminal, explain when we use these networking Command?
    Man, tcpdump, iwconfig, dig, tracepath
12. How to differentiate between Hub & Switch in Cisco Packet Tracer
13. Socket Programming- make a multithreaded socket server.
14. How to make Client Server Communication using Socket Programming?
15. Open Linux Terminal, explain when we use these networking Command?
    ifconfig, host, traceroute, mtr, ssh
16. Socket Programming- socket server with multiple Client.
17. Write a program to implement Affine Cipher.
18. Implement Bus Topology using Cisco Packet Tracer.
19. Implement Ring Topology using Cisco Packet Tracer.
20. Implement Mesh Topology using Cisco Packet Tracer.
21. Implement Star Topology using Cisco Packet Tracer.
22. Implement Hybrid Topology using Cisco Packet Tracer.
23. Implement Tree Topology using Cisco Packet Tracer.
24. Socket Programming- Sending and Receiving Data.
25. How to Use the FTP Command to Copy Files to/from a Server.
26. Create FTP Server using Cisco Packet Tracer.
27. Write a Program to Implement RSA Algorithm.

28. Write a Program to Implement Hill Cipher.
29. Write a Program to Implement Row Transposition Cipher.
30. Write a Program to Implement CRC Error Detection.

Semester - IV
Course Code: BCA/405/MN-4A
Course Title: Fuzzy Logic and its Applications (4 Credit)
Course Objectives:

- To develop the fundamental concepts such as fuzzy sets, operations and fuzzy relations.
- To lean about the fuzzification of scalar variables and the defuzzification of membership functions.
- To learn three different inference methods to design fuzzy rule based system.
- To develop fuzzy decision making by introducing some concepts and also Bayesian decision methods.
- To learn different fuzzy classification methods.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the basic ideas of fuzzy sets, operations and properties of fuzzy sets and also about fuzzy relations.
- Understand the basic features of membership functions, fuzzification process and defuzzification process.
- Design fuzzy rule based system.
- Know about combining fuzzy set theory with probability to handle random and non-random uncertainty, and the decision making process.
- Gain the knowledge about fuzzy C-Means clustering.

Theory: 60 Lectures

Unit-I: Classical sets: Operations and properties of classical sets, Mapping of classical sets to the functions. Fuzzy sets - Membership functions, Fuzzy set operations, Properties of fuzzy sets.

Unit-II: Classical and Fuzzy relations: Cartesian product, crisp relations-cardinality, operations and properties of crisp relations. Fuzzy relations-cardinality, operations, properties of fuzzy relations, fuzzy Cartesian product and composition, Fuzzy tolerance and equivalence relations, value assignments and other format of the composition operation.

Unit-III: Fuzzification and Defuzzification: Features of the membership functions, various forms, fuzzification, defuzzification to crisp sets, - cuts for fuzzy relations, Defuzzification to scalars. Fuzzy logic and approximate reasoning, other forms of the implication operation

Unit-IV: Fuzzy Systems: Natural language, Linguistic hedges, Fuzzy (Rule based) System, Aggregation of fuzzy rules, Graphical techniques of inference, Membership value assignments: Intuition, Inference, rank ordering, Fuzzy Associative memories

Unit-V: Fuzzy decision making: Fuzzy synthetic evaluation, Fuzzy ordering, Preference and consensus, Multi objective decision making, Fuzzy Bayesian, Decision method, Decision making under Fuzzy states and fuzzy actions

Unit-VI: Fuzzy Classification: Classification by equivalence relations-crisp relations, Fuzzy relations, Cluster analysis, Cluster validity, C-Means clustering, Hard C-Means clustering, Fuzzy C-Means algorithm, Classification metric, Hardening the Fuzzy C-Partition.

Text Books / Reference Books / Online Resources:

1. Timothy J.Ross - Fuzzy logic with engineering applications, 3rd edition, Wiley, 2010.

2. George J.KlirBo Yuan - Fuzzy sets and Fuzzy logic theory and Applications, PHI, New Delhi, 1995.
3. S.Rajasekaran, G.A.Vijayalakshmi - Neural Networks and Fuzzy logic and Genetic Algorithms,
4. Synthesis and Applications, PHI, New Delhi, 2003.

Semester - IV
Course Code: BCA/405/MN-4B
Course Title: Data Modeling and Design (4 Credit)
Course Objectives:
The purpose of this course is to provide a comprehensive introduction to database modeling and design implementing the Database Systems Development Life Cycle. The course will define and use essential database terms and concepts. The emphasis of the course is on the design and use of a relational database. The student will analyze complex business scenarios, and design and create models to support said scenario.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Describe the importance of information within an organization.
- Understand the Systems Development Life Cycle.
- Describe the process of modeling business requirements.
- Apply business concepts to a data model.
- Understand the concept of data normalization.
- Analyze user information requirements to develop an entity relationship model.
- Design and interpret conceptual data models.
- Develop a relational database design using an Entity Relationship Model.
- Identify and correct poor database designs.
- Describe Structured Query Language.
- Describe Data Warehouse databases.
- Understand data types and data objects, and define their use in Object-Oriented databases

Theory: 60 Lectures

Unit-I: Introduction to Database Technologies: Systems Development Life Cycle, Identify Business Requirements, Entity Relationship Data Modeling, Normalization, Advanced Data Modeling Concepts, Transform a Data Model into a Functional Database, Structured Query Language, Database Management Concepts, Data Warehousing, Object-Oriented Concepts.

Unit-II: Introduction to Database Concepts: Learn the history of data processing, Learn the history of databases, Learn essential database vocabulary, Understand the need to develop database systems, Understand the current position of database systems in organizational success.

Unit-III: Systems Development Life Cycle: Learn the Systems Development Life Cycle.

Unit-IV: Identifying Business Requirements: Describe the process of modeling business requirements, Determine entities among data requirements, Determine the attributes associated to entities, Determine the interrelatedness of entities / business matters of importance.

Unit-V: Entity Relationship Data Modeling: Understand the Data Modeling Process including scope definition and requirements, Define Entity, Attribute and Relationship, Convert a business rule to a ERD relationship, Illustrate the entities, attributes and relationships in an accurate ERD accurate graphic representation, Explain relationship's ordinality, Explain relationship's cardinality/degree, Verbalize the Entity Relationship Diagram's notation.

Unit-VI: Normalization: Define Normalization, Describe Normalization, Resolve a many – to – many relationship, Create an Entity Relationship Diagram in 3rd Normal Form.

Unit-VII: Advanced Data Modeling Concepts: Understand and model subtypes and supertypes, Understand and model hierarchical data, Understand and model recursive relationships, Understand and model historical data.

Unit-VIII: Transform a Data Model into a Functional Database: Distinguish between a conceptual model and a physical model, Define relational database terminology, Relate the conceptual design to a physical mode, Map entities and attributes, Define Primary Keys, Map relationships to Foreign Keys.

Unit-IX: Structured Query Language: Describe Structured Query Language, Demonstrate basic syntax in SQL (SELECT, FROM), Demonstrate selecting specific rows of data / projection (WHERE), Understand how to show a table design (DESCRIBE).

Unit-X: Data Warehousing: Describe Data Warehousing, Understand the difference between a transactional database and a data warehouse.

Unit-XI: Object Oriented Concepts: Define an object, Define an object orientated database.

Text Books / Reference Books / Online Resources:

1. Data Modeling Essential the Ultimate Reference, by Graeme C. Simsion, Dreamtech Press
2. Data Modeling Made Simple: A Practical Guide for Business and IT Professionals, Steve Hoberman, Originally Published.
3. Data Modeling: A Beginner's Guide, Andy Oppel, Originally Published.

Semester - IV
Course Code: BCA/405/MN-4C
Course Title: Cyber Security Techniques (4 Credit)
Course Objectives:
To prepare students with the technical knowledge and skills needed to protect and defend computer systems and networks. To develop graduates that can plan, implement, and monitor cyber security mechanisms to help ensure the protection of information technology assets.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Protect and defend computer systems and networks from cybersecurity attacks. ...
- Diagnose and investigate cybersecurity events or crimes related to computer systems and digital evidence.
- Effectively communicate in a professional setting to address information security issues.

Theory: 60 Lectures

Unit-I: Information Security Basics to Policy, Cybercrime and different modes of attacks, Intrusion detection system IT Assets and Wireless Security

Unit-II: Cyber Security Assurance Framework, Desktop Security and Malware, E-Commerce and Web-Application Security, Social Engineering.

Unit-III: Cyber Security Risk Management, Computer Forensics Fundamentals and Collection of Digital Evidence, Cyber Security Initiatives in India, Cyber Security Strategies and Policies.

Unit-IV: Network Security Threats, Network Security Technologies, Network Security – Controls and Best Practices, Network Security (Physical and Environment Security).

Text Books / Reference Books / Online Resources:

1. Cyber Security : Learn All the Essentials and Basic Ways to Avoid Cyber Risk for Your Business (Cybersecurity Guide for Beginners), Ben Chan.
2. Cybersecurity for Dummies, Joseph Steinberg.
3. Cyber Security, Notion Press.

Semester - IV
Course Code: BCA/405/MN-4D
Course Title: Software Designing (4 Credit)
Course Objectives:
During this course, students will:

- Discover how architecture influences (and is influenced by) technical environments, project lifecycles, business profiles and practices.
- Explore how various software quality attributes (i.e., reliability, usability, compatibility, interoperability, etc.) can influence the architecture and design of a software product.
- Leverage appropriate software design patterns, interfaces, and practices for optimizing quality of a software product through its carefully planned architecture and design.
- Use Agile development practices and the Unified Modeling Language (UML) in the development of a semester software design project.
- Architect and design a software product either for a mobile, cloud, embedded, or other type of software product.
- Investigate how to scale software systems, using evolutionary software development methodologies such as DevOps, software product lines, and/or deployment pipelines.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Develop the architectural, interface and component design elements for a comprehensive software design that meets specified design requirements.
- Collaborate with other students in a team setting using industry representative team collaboration methodologies to complete the software design.
- Document a software design using Unified Modeling Language (UML) architectural views and design constructs (i.e., sequence diagrams, deployment diagrams, timing diagrams, state transition diagrams, etc.).
- Select standard quality attributes to influence a product's software design.
- Architect /design a software product for a mobile, cloud, or embedded application
- Select common software design patterns used to develop software designs that meet specified requirements and quality attributes.

Theory: 60 Lectures

Unit-I: Introduction to Software Design: Describe software design engineering activities. • Compare and contrast software architecture and detailed design. • Evaluate the role of the SDLC Design phase in the production of high quality, sustainable software.

Unit-II: Understanding Software Architecture: Explain the importance of software architectural design. • Compare and contrast different architectural views to capture a software design.

Unit-III: Software Quality Attributes: Describe common software quality attributes. • Participate in a software quality attribute workshop (QAW) to select the quality attributes most important in a software design.

Unit-IV: Software Design Patterns and Interface Design: Connect software architectural elements to detailed design elements. • Describe basic strategies of using object-oriented design techniques. • Compare and contrast common

software design patterns. • Develop software interface design solutions.

Unit-V: Documenting a Software Architecture and Design: Describe how graphical models and UML are used to capture the design of a software system and its behavior • Capture a software design in a UML modeling tool.

Unit-VI: Scalable Software Architecture and Design: Incorporate open-source or reused software into software design. • Develop a plan to scale a software system, using evolutionary software development methodologies such as DevOps, software product lines, and/or deployment pipelines.

Unit-VII: Evaluating a Software Architecture and Design: Evaluate a software architecture for a given set of attribute-specific measures for a software system. • Conduct a software architecture review with stakeholders and an evaluation team. • Map architectural and design approaches to prioritized quality attributes. • Identify sensitivity points and tradeoff points of achieving specified quality attributes.

Unit-VIII: Conducting a Software Design Review: Prepare a software design presentation. • Conduct a software design review with a panel of subject matter experts. • Disposition any resulting actions from the software design review.

Text Books / Reference Books / Online Resources:

1. Software Design, David Budgen, Pearson Education India.
2. Software Architecture by Subodh Prasad, S.K. Kataria & Sons

Semester - IV
Course Code: BCA/407/INT-2
Course Title: Summer Internship (4 Credit)
Course Objectives:
Acquire on job the skills, knowledge, and attitude, which are requisite to constitute a professional identity. Demonstrate professional values and ethical standards**.**
Learning Outcomes:
After the completion of this course, the students will be able to:

- Student is able to construct the company profile by compiling the brief history, management structure, products / services offered, key achievements and market performance for his / her organization of internship.
- For his / her organization of internship, the student is able to assess its Strengths, Weaknesses, Opportunities and Threats (SWOT).
- Student is able to determine the challenges and future potential for his / her internship organization in particular and the sector in general.
- Student is able to test the theoretical learning in practical situations by accomplishing the tasks assigned during the internship period.
- Student is able to apply various soft skills such as time management, positive attitude and communication skills during performance of the tasks assigned in internship organization.
- Student is able to analyze the functioning of internship organization and recommend changes for improvement in processes.

Guidelines for Summer Training Report Writing

Viva Voce on Summer Training is an important part of BCA course curriculum. It providesan opportunity to BCA students to write a summer training report on latest trends/technology related to software project. Following are guidelines for summer training report writing and assessment:

General Instructions

- Summer training report should not be less than 15 pages.
- Proper guidelines to be followed for preparation of summer training report.
- Proper dress code is mandatory for presenting and attending summer training PPT presentations.
- Attendance is compulsory for all students.
- If a student is absent for his presentation as per schedule, he/she must have assessed later on with reducedweightage in the presentation assessment.
- Always prepare a draft report first and print it out.
- Read it yourself first and correct any typographical or grammatical errors.
- **One copy** of final summer training report must be submitted as a **spiraled report** to the coordinator.

Main Components of a Report

- **Cover page**
  As per the given format.
- **Abstract**
  Abstract should be a narration of Problem Definition, Methodology used in third person's language not exceeding 200 words.
- **Acknowledgement and declaration**

The Acknowledgement shall be brief and should not exceed one page. The students' signature shall bemade at the bottom end above their name typed in capital.

- **Certificate**
  Attach a photocopy of a certificate issued from where training has been done.
- **Table of contents/Index page**
  The table of contents should list all material following it. The abstract, acknowledgement will be included in table of contents but the page numbers in lower case Roman letters are to be accounted forthem. Pages starting with Introduction (Immediate after the Table of contents) should be consecutivelynumbered using Arabic Numerals.
- **Main Text with conclusions**
  This includes various sections for Summer Training Report.
- **References**

Typing Instructions for Summer Training Report

- **Specification for Fonts**
  - **Font Face: Times new Roman**
  - **Font Size:** As per following preview
    - **Headings (Size 16 Bold)**
    - **Sub-Heading (Size 14 Bold and Italic)**

- **Contents (Size 12 Normal)**
  - **Line spacing: 1.5**
  - **Text Alignment: Both left and right justified**

- Page Dimensions: Standard A4 size (297mm x 210mm)
- Margins:
  - **Top margin: 0.75”**
  - **Bottom margin: 0.75”**
  - **Left margin: 1”**
  - **Right margin: 0.75”**

Footer: Page number should be bottom centered.

- Sections should be numbered as for example,**1. Introduction** etc.
- Subsections should be numbered as for example,***3.1 Simulation Tool*** etc.
- Paragraphs and sentences should be short.
- Start of a paragraph should not be intended, rather, give one-line space between two paragraphs.
- A sub heading at the bottom of a page must have at least two full lines below it or else it should becarried over to the next page.

- The last word of any page should not be split using a hyphen.

References

- References should be in IEEE format, i.e.,
  - Author name(s), title of paper/book, or if journal, its name, volume, issues, page number,publisher, year of publication
- Book titles must be in capitals
- Reference numbers should be marked liberally inside the text of the report-
  - eg., ……as given in [3]

References should either be in chronological order or in the order in which they appear in thetext.

Semester - V
Course Code : BCA-501/MJC-9
Course Title : Computer Graphics and Multimedia (4 Credit)
Course Objectives:
The main objective of this module is to introduce to the students the concepts of computer graphics. It starts with an overview of interactive computer graphics, two dimensional system and mapping, then it presents the most important drawing algorithm, two-dimensional transformation; Clipping, filling and an introduction to 3-D graphics.
Learning Outcomes:
At the end of the course, the students will be able to:

- Understand how to generate line, circle and ellipse also how to create 2D object and various transformation techniques.
- Understand various 3D Transformation techniques.
- Understand multimedia compression techniques and applications.

Theory: 60 Lectures

Unit-I: Introduction to computer graphics & graphics systems: Overview of computer graphics, representing pictures, preparing, presenting & interacting with pictures for presentations; Visualization & image processing; RGB color model, direct coding, lookup table;

Unit-II: Devices: storage tube graphics display, Raster scan display, 3D viewing devices, Plotters, printers, digitizers, Light pens etc.; Active & Passive graphics devices; Computer graphics software.

Unit-III: Scan conversion Points & lines: Line drawing algorithms; DDA algorithm, Bresenham's line algorithm,

Unit-IV: Scan conversion -2: Circle generation algorithm; Ellipse generating algorithm; scan line polygon, fill algorithm, boundary fill algorithm, flood fill algorithm.

Unit-V: 2D transformation Basic transformations: translation, rotation, scaling ; Matrix representations & homogeneous coordinates, transformations between coordinate systems; reflection shear; Transformation of points, lines , parallel lines, intersecting lines.

Unit-VI: 2D Viewing: Viewing pipeline, Window to viewport Co-ordinate transformation, clipping operations, point clipping, line clipping, clipping circles, polygons & ellipse.

Unit-VII: 3D transformation & viewing 3D transformations: translation, rotation, scaling & other transformations. Rotation about an arbitrary axis in space, reflection through an arbitrary plane; general parallel projection transformation; clipping, viewport clipping, 3D viewing.

Unit-VIII: Curves: Curve representation, surfaces, designs, Bezier curves, B-spline curves, end conditions for periodic B-spline curves, rational B-spline curves.

Unit-IX: Hidden surfaces Depth comparison: Z-buffer algorithm, Back face detection, BSP tree method, the Printer's algorithm, scan-line algorithm; Hidden line elimination, wire frame methods, fractal - geometry.

Unit-X: Color & shading models: Light & color model; interpolative shading model; Texture;

Unit-XI: Graphics Languages: GKS, PHIGS

Unit-XII: Multimedia: Introduction to multimedia hardware, Networking, software applications, Environment, CDROM, WORM Optical Drives, Flat panel Displays, Non Temporal Media-Text, Hypertext, Images, Image Operations, CCD Cameras, Scanners, Frame Grabbers, Formats, Audio Digital Audio, Wave Files, Music, MIDI, Graphics Animation-Tweeking, Morphing, Simulating Acceleration, Motion specification, Video-Analog video: Operations, Digital Video, Compression, MPEG, JPEG, Operations, Multimedia Authoring Systems.

Reference Books:

1. D. Hearn, Baker: Computer Graphics, Prentice Hall of India 2008.
2. Foley, Vandam, Feiner, Huges, "Computer Graphics: Principles & Practice", Pearson Education, second edition 2003.

3. Procedural Elements of Computer Graphics – D. F. Rogers McGraw Hill
4. Principles of Interactive Graphics – Newman & Sproull McGraw Hill

Practical

1. Write a C program to draw a line between 10, 5 and 20, 300 using appropriate line drawing algorithm.
2. Write a C program to draw a line between 100, 5 and 120, 10 using appropriate line drawing algorithm.
3. Write a C program to draw a line between 20, 15 and 20, 30 using appropriate line drawing algorithm.
4. Write a C program to draw a line between 60, 55 and 50, 65 using appropriate line drawing algorithm.
5. Write a C program to draw a line between 70, 25 and 80, 25 using appropriate line drawing algorithm.
6. Write a C program to draw Circle using Bresenham's algorithm with circle center at 165,172 and radius 8.
7. Write a C program to draw a circle with center 15,20 and radius 12 using Bresenham's algorithm.
8. Write a C program to draw Ellipse using Bresenham's algorithm.
9. Write a program to apply scaling and rotation to a line.
10. Write a program to apply scaling and shearing to 2-Dimensional shapes.
11. Write a program to apply translation 10 pixel towards x axis and 15 pixel towards y axis to triangle (10,10; 30,15; 15,25).
12. Write a program to apply translation and rotation to 2-Dimensional shapes.
13. Write a program to apply translation and shearing to 2-Dimensional shapes.
14. Write a program to apply reflection W. R. To. X axis to 2-Dimensional shapes.
15. Write a program to apply translation and reflection W. R. To. Y axis to 2-Dimensional shapes.
16. Write a program to draw a circle with center 15, 17 and radius 10, and fill it with red color using boundary fill algorithm.
17. Write a program to draw a circle with center 10, 17 and radius 8, and fill it with boundary color using boundary fill algorithm.
18. Write a program to draw a circle with center 100,100 and radius 11, and fill it with red color using any fill algorithm.
19. Write a program to draw a circle with center 20, 20 and radius 14, and fill it with user inputted color using flood fill algorithm.
20. Write a program to draw a circle with center 15, 20 and radius 12, and fill it with red color using flood fill algorithm.

Semester - V
Course Code : BCA/502/MJC-10
Course Title : Database Management Systems (4 Credit)
Course Objectives:
The objective of the course is to present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS.
Learning Outcomes:
At the end of the course, the students will be able to:

- *Understand* the basic concepts and the applications of database systems.
- *Master* the basics of SQL and construct queries using SQL.
- *Understand* the relational database design principles.
- *Familiar* with the basic issues of transaction processing and concurrency control.
- *Familiar* with database storage structures and access techniques

Theory: 60 Lectures

Unit-I: Database Management System Concepts: Introduction, Significance of Database, Database System Applications; Data Independence; Data Modeling for a Database; Entities and their Attributes, Entities,

Attributes, Relationships and Relationships Types, Advantages and Disadvantages of Database Management System, DBMS Vs RDBMS.

Unit-II: Database System Architecture: Three Level Architecture of DBMS, The External Level or Subschema, The Conceptual Level or Conceptual Schema, The Internal Level or Physical Schema, Mapping; MySQL Architecture; SQL Server 2000 and above Architecture; Oracle Architecture; Database Management System Facilities, Data Definition Language, Data Manipulation Language; Database Management System Structure, Database Manager, Database Administrator, Data Dictionary; Distributed Processing, Information and Communications Technology System (ICT), Client / Server Architecture

Unit-III: Database Models and Implementation: Data Model and Types of Data Model, Relational Data Model, Hierarchical Model, Network Data Model, Object/Relational Model, Object-Oriented Model; Entity-Relationship Model, Modeling using E-R Diagrams, Notation used in E-R Model, Relationships and Relationship Types; Associative Database Model

Unit-IV: File Organization for Conventional DBMS: Storage Devices and its Characteristics, Magnetic Disks, Physical Characteristics of Disks, Performance Measures of Disks, Optimization of Disk-Block Access; File Organization, Fixed-Length Records, Variable-Length Records, Organization of records in files; Sequential file Organization; Indexed Sequential Access Method (ISAM); Virtual Storage Access Method (VSAM)

Unit-V: An Introduction to RDBMS: An informal look at the relational model; Relational Database Management System; RDBMS Properties, The Entity-Relationship Model; Overview of Relational Query Optimization; System Catalog in a Relational DBMS, Information Stored in the System Catalog, How Catalogs are Stored

Unit-VI: Relational Algebra: Basic Operations, Union (U), Difference (-), Intersection ($\cap$), Cartesian Product (x); Additional Relational Algebraic Operations, Projection (π), Selection (σ), JOIN ( ), Division ($\div$)

Unit-VII: Relational Calculus: Tuple Relational Calculus, Semantics of TRC Queries, Examples of TRC Queries; Domain Relational Calculus; Relational ALGEBRA vs Relational CALCULUS

Unit-VIII: Normalization: Functional Dependency; Anomalies in a Database; Properties of Normalized Relations; First Normalization; Second Normal Form Relation; Third Normal Form; Boyce-Codd Normal Form (BNCF); Fourth and Fifth Normal Form

Unit-IX: SQL: Categories of SQL Commands; Data Definition; Data Manipulation Statements, SELECT - The Basic Form, Subqueries, Functions, GROUP BY Feature, Updating the Database, Data Definition Facilities, Views; Embedded SQL *, Declaring Variables and Exceptions, Embedding SQL Statements, Transaction Processing, Consistency and Isolation, Atomicity and Durability

Unit-X: Indexing and Hashing: Introduction, Overview, Primary Secondary Multi level, Dense and Space Index.

Unit-XI: NoSQL: Overview and History of NoSQL; Definition of 4 types of NoSQL Database and NoSQL data store; NoSQL data architecture patterns; NoSQL to manage Big Data; Comparison of relation database to new NoSQL stores, MongoDB, Redis, Cassandera, Neo4j use and deployment.

Reference Books:

1. An Introduction to Database Systems, Vol.I & II – C.J. Date, Addison Wesley.
2. Database System Concepts, 3rd edn. – Corth & Siberschatz, T.M.H
3. Principles of Database Systems, 2nd edn. – J.D. Ullman, Galgotia
4. Fundamentals of Database Systems, R. Elmasri, S.B. Navathe, 6th E, PE, 2010
5. Database Management Systems, R. Ramakrishanan, J. Gehrke, 3rd E, MGH, 2002

6. Fundamentals of SQL – Evan Bayross.
7. NoSQL FOR DUMMIES – Adam Fowler Wiley Publication

Practical

1. EMPLOYEES (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id)
Create the above table and insert 10 records into it. Now perform the following queries.
(a) Find out the employee id, names, salaries of all the employees
(b) List out the employees who works under manager 100
(c) Find the names of the employees who have a salary greater than or equal to 4800
(d) List out the employees whose last name is 'AUSTIN'
(e) Find the names of the employees who works in departments 60,70 and 80
(f) Display the unique Manager_Id.

2. Create Client_master with the following fields(ClientNO, Name, Address, City, State, bal_due)
Create the above table and insert 10 records into it. Now perform the following queries.
(a) Find the names of clients whose bal_due> 5000 .
(b) Change the bal_due of ClientNO " C123" to Rs. 5100
(c) Change the name of Client_master to Client12 .
(d) Display the bal_due heading as "BALANCE"
(e) Display data of all clients for the following fields and sequence(state, city, name clientno)

3. Create Teacher table with the following fields(Name, DeptNo, Date of joining, DeptName, Location, Salary)
(a) Insert ten records
(b) Give Increment of 25% salary for Mathematics Department .
(c) Perform Rollback command
(d) Give Increment of 15% salary for Commerce Department
(e) Perform commit command

4. Create Sales table with the following fields(Sales No, Salesname, Branch, Salesamount, DOB)
(a) Insert five records
(b) Calculate total salesamount in each branch
(c) Calculate average salesamount in each branch .
(d) Display all the salesmen, DOB who are born in the month of December as day in character format i.e. 21-Dec-09.
(e) Display the name and DOB of salesman in alphabetical order of the month.

5. Create an Emp table with the following fields:
EmpNo, EmpName, Job,Basic, DA, HRA,PF, GrossPay, NetPay)
(Calculate DA as 30% of Basic and HRA as 40% of Basic)
(a) Insert Five Records and calculate GrossPay and NetPay.
(b) Display the employees whose Basic is lowest in each department .
(c) If NetPay is less than
(d) Display the employees whose GrossPay lies between 10,000 & 20,000
(e) Display all the employees who earn maximum salary .

6. Employee Database
An Enterprise wishes to maintain a database to automate its operations. Enterprise is divided into certain departments and each department consists of employees. The following two tables describes the automation schemas

Dept (deptno, dname, loc)
Emp (empno, ename, job, mgr, hiredate, sal, comm, deptno)
(a) Update the employee salary by 15%, whose experience is greater than 10 years.
(b) Delete the employees, who completed 30 years of service.
(c) Display the manager who is having maximum number of employees working under him? (d) Create a view, which contain employee names and their manager

7. Using Employee Database perform the following queries
(a) Determine the names of employee, who earn more than their managers.
(b) Determine the names of employees, who take highest salary in their departments.
(c) Determine the employees, who are located at the same place.
(d) Determine the employees, whose total salary is like the minimum Salary of any department.
(e) Determine the department which does not contain any employees.

8. Consider the following tables namely "DEPARTMENTS" and "EMPLOYEES" Their schemas are as follows,
Departments (dept _no , dept_ name , dept_location);
Employees (emp_id , emp_name , emp_salary,dept_no);
(a) Develop a query to grant all privileges of employees table into departments table
(b) Develop a query to grant some privileges of employees table into departments table
(c) Develop a query to revoke all privileges of employees table from departments table
(d) Develop a query to revoke some privileges of employees table from departments table (e) Write a query to implement the save point.

9. Using the tables "DEPARTMENTS" and "EMPLOYEES" perform the following queries
(a) Display the employee details, departments that the departments are same in both the emp and dept.
(b) Display the employee name and Department name by implementing a left outer join.
(c) Display the employee name and Department name by implementing a right outer join. (d) Display the details of those who draw the salary greater than the average salary.

10. Consider the following relational schema for the Office of the Controller of Examinations Application.
Student (Rollno, Name, Dob, Gender, Doa, Bcode);
Implement a check constraint for (Gender, Date of Admission)
Branch (Bcode, Bname, Dno);
Department (Dno, Dname);
Course (Ccode, Cname, Credits, Dno);
Branch_Course (Bcode, Ccode, Semester);
Enrolls (Rollno, Ccode, Sess, Grade);
For Example, SESS can take values 'APRIL 2013', 'NOV 2013'
Implement a check constraint for grade Value Set ('S', 'A', 'B', 'C', 'D', 'E', 'U');
Students are admitted to Branches and they are offered by Departments. A branch is offered by only one department.
Each branch has a set of Courses (Subjects). Each student must enroll during a semester. Courses are offered by Departments. A course is offered only by one department. If a student is unsuccessful in a course he/she must enroll for the course during next session. A student has successfully completed a course if the grade obtained by is from the list (A, B, C, D, and E).
A student is unsuccessful if he/she have grade 'U' in a course.
Primary Keys are underlined.
(a) Develop a SQL query to list details of Departments that offer more than 3 branches.
(b) Develop a SQL query to list the details of Departments that offer more than 6 courses.
(c) Develop a SQL query to list the details of courses that are common for more than 3 branches.
(d) Develop a SQL query to list students who got 'S' in more than 2 courses during single enrollment.

(e) Create a view that will keep track of the roll number, name and number of courses, a student has completed successfully.

Semester – V
Course Code : BCA/503/MJC-11
Course Title : Microprocessor and System Software (4 Credit)
Course Objectives:
Students will be able to.

- Outline the history of computing devices.
- Describe the architecture of 8085 microprocessors.
- Develop programs for microprocessor and microcontrollers
- Compare microprocessors and microcontrollers

Learning Outcomes:
After the completion of this course, the students will be able to:

- Define the history of microprocessors.
- Describe the architectures of 8085 microprocessors.
- Draw timing diagram.
- Write programs using 8085.
- Distinguish between the different modules of operation of microprocessors.
- Interface peripherals to 8085.

Theory: 60 Lectures

Unit-I: Introduction of Microcomputer System: CPU, I/O devices, clock, memory, bussed architecture, tristate logic, address bus, data bus and control bus.

Unit-II: Semiconductor Memories: Development of semiconductor memory, internal structure and decoding, memory read and write timing diagrams, MROM, ROM, EPROM, EEPROM, DRAM.

Unit-III: Architecture of 8-bit Microprocessor: Intel 8085 microprocessor, Pin description and internal architecture.

Unit-IV: Operation and Control of Microprocessor: Timing and control unit, op-code fetch machine cycle, memory read/write machine cycles, I/O read/write machine cycles, interrupt acknowledge machine cycle, state transition diagram.

Unit-V: Instruction Set: Addressing modes, Data transfer, arithmetic, logical, branch, stack and machine control groups of instruction set, macro RTLand micro RTL flow chart of few typical instructions, unspecified flags and instructions.

Unit-VI: Assembly Language Programming: Assembler directives, simple examples; Subroutines, parameter, passing to subroutine.

Unit-VII: Interfacing: Interfacing of memory chips, address allocation technique and decoding; Interfacing of I/O devices, LEDs and toggle-switches as examples, memory mapped and isolated I/O structure; Input/output techniques: CPU initiated unconditional and conditional I/O transfer, device initiated interrupt I/O transfer.

Unit-VIII: Interrupts: Interrupt structure of 8085 microprocessors, processing of vectored and non-vectored interrupts, latency time and response time; Handling multiple interrupts

Unit-IX: Programmable Peripheral Interface: Intel 8255, pin configuration, internal structure of a port bit, modes of operation, bit SET/RESET feature, programming, DC and DAC chips and their interfacing.

Unit-X: Programmable Interval Timer: Intel 8253, pin configuration, internal block diagram of counter and modes of operation, counter read methods, programming, READ-BACK command of Intel 8254.

Unit-XI: System Programming: Assemblers-Assembler directives, design of two-pass assembler, Microprocessors-Macro definition and expansion, conditional macro expansion, macro defining macros, design of microprocessor (without macro defining macros).

Unit-XII: Loader and linker: Loader and linker -relocation and linking, dynamic linking and overlaying programs, Compilers-phases in compilation process and operations done in each phase.

Reference Books:

1. Microprocessors: A. P. Mathur
2. Microcomputer systems -8086/8088 family: Liu and Gibson
3. Programming the 80286, 80386 Computers: B. B. Brey
4. Microprocessors and Interfacing: Hall
5. Microprocessor and Interfacing-Programming and Hardware, Hall D. V., 2nd Ed., Tata McGraw-Hill Publishing Company Limited, 2008
6. Microprocessor Architecture, Programming and Applications, Gaonkar R. S., 5th Ed., Penram International, 2007.
7. Microprocessor Systems- Hardware, Software and Programming, Stewart J, Prentice Hall International Edition,1990
8. Microprocessors and Programmed Logic, Short K. L., 2nd Ed., Pearson Education, 2008.

Practical

1. Write an assembly language program to perform addition of two 8-bit numbers using 8085 instruction set.
2. Write an assembly language program for subtracting of two 8-bit numbers by using-8085.
3. Write an assembly language program to find the 2's compliment of an 8 bit decimal numbers by using-8085.
4. Write an assembly language program to find the larger of the two numbers (04H and 08H) by using-8085.
5. To write an assembly language program to arrange 3 numbers in descending order by using-8085.
6. Write an assembly language program to multiply two 8-bit numbers by using-8085 micro-processor kit.
7. Write an assembly language program to find the largest number in array of data using 8085.
8. Write an assembly language program to find the smallest number in array of data using 8085.
9. Write an assembly language program to arrange an array of data in ascending order by using-8085.
10. Write an assembly language program to convert two BCD numbers in memory to the equivalent Hexa Decimal number using 8085 instruction set.
11. Write an assembly language program to convert two Hexa Decimal numbers in memory to the equivalent BCD number using 8085 instruction set.
12. Write an assembly language program to check the 4^{th} bit of 8-numbers.

Semester - V
Course Code : BCA/504/MJC-12
Course Title : Introduction to Internet of Things (4 Credit)
Course Objectives:

- To introduce the concepts and techniques associated with Wireless Cellular Communication systems.

- To familiarize with state of art standards used in wireless cellular systems.

Learning Outcomes:

- Explain the Classification of mobile communication systems
- Analyse the radio channel characteristics and the cellular principle
- Analyse the measures to increase the capacity in GSM systems- sectorization and Spatial Filtering for Interference Reduction

Ability to analyse improved data services in cellular communication

Theory: 60 Lectures

Unit-I: IOT: What is the IoT and why is it important? Elements of an IoT ecosystem, Technology drivers, Business drivers, Trends and implications, Overview of Governance, Privacy and Security Issues.

Unit-II: IOT PROTOCOLS: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Issues with IoT Standardization – Unified Data Standards – Protocols – IEEE802.15.4–BACNet Protocol– Modbus – KNX – Zigbee– Network layer – APS layer – Security

Unit-II: IOT ARCHITECTURE: IoT Open source architecture (OIC)- OIC Architecture & Design principles- IoT Devices and deployment models- IoTivity : An Open source IoT stack - Overview- IoTivity stack architecture- Resource model and Abstraction.

Unit-IV: WEB OF THINGS: Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence.

Unit-V: IOT APPLICATIONS: IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications. Study of existing IoT platforms /middleware, IoT- A, Hydra, AI based IoT Applications etc.

Reference Books:

1. Honbo Zhou, “The Internet of Things in the Cloud: A Middleware Perspective”, CRC Press,2012.
2. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), “Architecting the Internet of Things”, Springer, 2011.
3. David Easley and Jon Kleinberg, “Networks, Crowds, and Markets: Reasoning About a Highly Connected World”, Cambridge University Press, 2010.
4. Olivier Hersent, David Boswarthick, Omar Elloumi , “The Internet of Things – Key applicationsand Protocols”, Wiley, 2012.
5. Vijay Madisetti and ArshdeepBahga, “Internet of Things (A Hands-on-Approach)”,1st Edition, VPT, 2014
6. Francis daCosta, “Rethinking the Internet of Things: A Scalable Approach to ConnectingEverything”, 1st Edition, Apress Publications, 2013
7. CunoPfister, Getting Started with the Internet of Things, O‟Reilly Media, 2011, ISBN: 978-1- 4493-9357-1

Semester - V

Course Code: BCA/505/MN-5A

Course Title: Knowledge Engineering and Expert Systems (4 Credit)

Course Objectives:

- To provide an overview of the Expert System.
- To introduce students about insights of the several topics of Expert System such as – building an expert system and knowledge engineering.
- To provide comprehensive details about various Evaluation methods of the expert system.
- To provide implementation insight about the topics covered in the course.

Learning Outcomes:

After the completion of this course, the students will be able to:

- Explain and describe the concepts central to the creation of knowledge bases and expert systems. Knowledgeable about the tools and the processes used for the creation of an expert system.
- Know the methods used to evaluate the performance of an expert system.
- Conduct an in-depth examination of an existing expert system with an emphasis on basic methods of creating a knowledge base.
- Examine the properties of existing systems in a case-study manner, comparing differing approaches.

Theory: 60 Lectures

Unit-I: Introduction: The nature of Expert Systems. Types of applications of Expert Systems; the relationship of Expert Systems to Artificial Intelligence and Knowledge-Based Systems. The nature of expertise. Distinguishing features of Expert Systems. Benefits of using an Expert System, choosing an application.

Unit-II: Theoretical Foundations: What an expert system is; how it works and how it is built—basic forms of inference: abduction; deduction; induction.

Unit-III: The representation and manipulation of knowledge in a computer: Rule-based representations (with backward and forward reasoning); logic-based representations (with resolution refutation); taxonomies; meronomies; frames (with inheritance and exceptions); semantic and partitioned nets (query handling).

Unit-IV: Basic components of an expert system: Generation of explanations. Handling of uncertainties. Truth Maintenance Systems. Expert System Architectures. An analysis of some classic expert systems. Limitations of first-generation expert systems. Deep expert systems. Co-operating expert systems and the blackboard model.

Unit-V: Building Expert Systems: Methodologies for building expert systems: knowledge acquisition and elicitation; formalisation; representation and evaluation. Knowledge Engineering tools, Case Study

Reference Books:

1. P Jackson, Introduction to Expert Systems, Addison Wesley, 1990 (2nd Edition)
2. Elaine Rich, Kevin Knight, Artificial Intelligence, McGraw-Hill, Inc, 1991 (2nd Edition)
3. Jackson. Jean-Louis Lauriere, Problem Solving and Artificial Intelligence, Prentice Hall, 1990

Semester - V
Course Code: BCA/505/MN-5B
Course Title: Statistic for Data Science (4 Credit)
Course Objectives:

- To provide an overview of the Statistic for Data Science.
- To introduce students about insights of the several topics of Statistic for Data Science.
- To provide comprehensive details about various Evaluation methods of the Statistic for Data Science.
- To provide implementation insight about the topics covered in the course.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Build a classification model and interpret results. Understand the concept of hypothesis testing. Learn the intricacies of logistic regression, evaluate its outputs, and comprehend how a link function works.
- Proficient in the statistical analysis of data and the use of computation tools for data analysis.
- Apply statistical and computational tools to applied problems, and clearly communicate the results in both written reports and oral presentations.

Theory: 60 Lectures

Unit-I: Introduction: What is Big Data, Difference between Big Data and Smart Data, What is Data Science?

Unit-II: Basic Probability Theory: Concepts of Events, Probability of an Event, Probability Rules, Independence.

Unit-III: Bayesian Probability: Prior Probability, Posterior Probability, Bayes Rule, Applications in Data Science (Classification).

Unit-IV: Random Variable: Random Variable, Discrete and Continuous, Cumulative density function, Multivariate Random Variable, Joint and Conditional, Independence, Common Discrete and Continuous Distributions.

Unit-V: Expectation and Variance: Concept of Expectation and Variance, Data-based Estimators (Sample mean and Sample Covariance), Generating Multivariate Random Variables.

Unit-VI: Linear Regression: Linear Models, Mean Square Error (MSE), Likelihood Function, Parameter Estimation (MLE), Over-fitting/Under-fitting.

Unit-VII: Cross Validation: What is Cross Validation, different methods, practical example.

Unit-VIII: Supervised and Unsupervised Learning: Labeled Data and Unlabeled Data, Clustering.

Unit-IX: Gaussian Distribution: Definition of Gaussian Distribution, Standard Gaussian Distribution, Visualization, Sample Mean and Covariance Matrix - MLE Estimation, Application - Modeling Noisy Data.

Unit-X: Hypothesis Testing: The hypothesis-testing framework, Parametric Testing, Non-Parametric Testing, Multiple Testing.

Unit-XI: Time-series/Sequential Data Science: What is Random Process, Independent identically-distributed sequences, Gaussian Process, Random Walks, Application: Time-dependent Prediction.

Unit-XII: Processing and Summarizing Data: Real-world Big Data Processing, Practical Examples, Practical Statistical Analysis.

Reference Books:

1. Carlos Fernandez-Granda, Probability and Statistics for Data Science, Center for Data Science in NYU, 2017, available online.
2. Avrim Blum, John Hopcroft, and Ravi Kannan, Foundations of Data Science, Cambridge University Press, March 2020, available online.
3. James D. Miller, STATISTICS FOR DATA SCIENCE,Packt Publishing, 2019.
4. Davy Cielen, Arno D. B. Meysman, and Mohamed Ali, Introducing Data Science, Manning Publication, 2016, available online.
5. Michael J. Evans and Jeffrey S. Rosenthal, Probability and Statistics The Science of Uncertainty, University of Toronto, 2009.
6. Joseph C. Watkins, An Introduction to the Science of Statistics: From Theory to Implementation, Preliminary Edition.

Semester - VI
Course Code: BCA/605/MN-6C
Course Title: Cryptography and Web Security (4 Credit)
Course Objectives:

- To understand basics of Cryptography and Network Security.

- To be able to secure a message over insecure channel by various means.
- To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
- To understand various protocols for network security to protect against the threats in the networks.

Learning Outcomes:

After the completion of this course, the students will be able to:

- Provide security of the data over the network.
- Do research in the emerging areas of cryptography and network security.
- Implement various networking protocols.
- Protect any network from the threats in the world.

Theory: 60 Lectures

Unit-I: Introduction: Security trends – Legal, Ethical and Professional Aspects of Security, Need for Security at Multiple levels, Security Policies – Model of network security – Security attacks, services and mechanisms – OSI security architecture – Classical encryption techniques: substitution techniques, transposition techniques, steganography- Foundations of modern cryptography: perfect security – information theory – product cryptosystem – cryptanalysis.

Unit-II: Symmetric Key Cryptography: MATHEMATICS OF SYMMETRIC KEY CRYPTOGRAPHY: Algebraic structures – Modular arithmetic-Euclid's algorithm- Congruence and matrices -Groups, Rings, Fields- Finite fields- SYMMETRIC KEY CIPHERS: SDES – Block cipher Principles of DES – Strength of DES – Differential and linear cryptanalysis – Block cipher design principles – Block cipher mode of operation – Evaluation criteria for AES – Advanced Encryption Standard – RC4 – Key distribution.

Unit-III: Public Key Cryptography: MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY: Primes – Primality Testing – Factorization – Euler's totient function, Fermat's and Euler's Theorem – Chinese Remainder Theorem – Exponentiation and logarithm – ASYMMETRIC KEY CIPHERS: RSA cryptosystem – Key distribution – Key management – Diffie Hellman key exchange -ElGamal cryptosystem – Elliptic curve arithmetic-Elliptic curve cryptography.

Unit-IV: IP Security and Key Management: IP Security: Architecture - Authentication header - Encapsulating security payloads - combining security associations - key management.

Unit-V: Security Practice and System Security: Electronic Mail security – PGP, S/MIME – IP security – Web Security – SYSTEM SECURITY: Intruders – Malicious software – viruses – Firewalls.

Unit-VI: Web and System Security: Web Security: Secure socket layer and transport layer security - secure electronic transaction (SET) - System Security: Intruders - Viruses and related threads - firewall design principals – trusted systems.

Reference Books:

1. William Stallings: Cryptography and Network Security- Principles And Practice, 5th Edition, Pearson/PHI, 2011
2. Charlie Kaufman, Radia Perlman and Mike Speciner: "Network Security – Private Communication in a Public World", 2nd Edition, Pearson/PHI, 2002.
3. Eric Maiwald: "Fundamentals of Network Security", 1st Edition, Dreamtech Press, 2003.
4. Whitman: "Principles of Information Security", 3rd Edition, Thomson, 2009.
5. Robert Bragg, Mark Rhodes: "Network Security: The complete reference", 1st Edition, TMH, 2004.
6. Buchmann: "Introduction to Cryptography", 2nd Edition, Springer, 2004

Semester - V

Course Code: BCA/505/MN-5D

Course Title: Advance Software Engineering (4 Credit)

Course Objectives:

- Understand object-oriented analysis and design basics.
- Be able to distinguish between problems and solutions and know how to map object oriented analysis to object oriented design.
- Employ fundamental design principles and patterns in the development of a software system and its supporting documents.
- Analyze architectural factors and use design mechanisms to select appropriate design technique and design patterns.
- Be introduced to the Service Oriented Architecture (SOA) and the Microservices.

Learning Outcomes:

After the completion of this course, the students will be able to:

- Critically analyze and evaluate the implications of software reuse.
- Understand full life-cycle design, implementation and testing of appliations built using the serivce-oriented methodology.
- Present software engineering concepts in a concise manner to an audience.
- Perform independed research on a topic and present it to the audience.

Theory: 60 Lectures

Unit-I: Introduction: FAQs about Software Engineering; Professional and Ethical Responsibility; Software Process: Models; Process Iteration, Specification, Software Design and Implementation; Verification & Validation; Software Evolution; Automated Process Support.

Unit-II: Software Project Management and Requirements: Project Management: Management Activities, Project Pl Software Project Management and Requirements Project Management: Management Activities, Project Planning, Project Scheduling, Risk Management; Software Requirements: Functional and Non-Functional Requirements, User Requirements, System Requirements, Requirements Document; Requirements Engineering Process: Feasibility Studies, Requirements Elicitation and Analysis, Requirements Validation, Requirements Management.

Unit-III: System Model, Software Prototyping and Specifications: System Models, Software Prototyping and Specifications System models: Context, Behavioural, Data, and Object models, CASE Workbenches; Software Prototyping: Prototyping in the Software Process, Rapid Prototyping Techniques, User Interface Prototyping; Specifications: Formal Specification in the Software Process, Interface Specification, Behavioural Specification.

Unit-IV: Architectural Design: Introduction: System Structuring; Control Models; Modular Decomposition; Domain-Specific Architectures; Distributed Systems Architectures: Multiprocessor Architectures; Client-Server Architectures, Distributed Object Architectures; CORBA (Common Object Request Broker Architecture).

Unit-V: Software Design: Object Oriented Design: Objects and Object Classes, Object-Oriented Design Process, Design Evolution; Real Time Software Design: Systems Design, Real-Time Executives, Monitoring and Control Systems, Data Acquisition Systems; Design with Reuse: Component-Based Development, Application Families, Design Patterns; User Interface Design: Principles, User Interaction, Information Presentation, User Support, Interface Evaluation.

Unit-VI: Verification and Validation and Testing: Verification and Validation (V & V): Static and Dynamic V & V, V & V Goals, V & V vs. Debugging, Software Inspections / Reviews, Clean-Room Software Development; Software Testing: Defect Testing, Integration Testing, Interface Testing, Object-Oriented Testing, Testing Workbenches

Unit-VII: Managing People: Introduction; Limits to Thinking; Memory Organization; Knowledge Modeling; Motivation; Group Working; Choosing and Keeping People; the People Capability Maturity Model.

Unit-VIII: Software Cost Estimation and Quality Management: **Software Cost Estimation:** Productivity, Estimation

Techniques, Algorithmic Cost Modelling, Project Duration and Staffing. Quality Management: Quality Assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics; Process Improvement: Process and Product Quality, Process Analysis and Modelling, Process Measurement, the SEI Process Maturity Model, and Process Classification.

Unit-IX: Evolution: Legacy Systems: Structures, Design, and Assessment; Software Change: Program Evolution Dynamics, Software Maintenance, Architectural Evolution; Software Re- Engineering: Source Code Translation, Reverse Engineering, Program Structure Improvement, Program Modularization, Data Re-Engineering; Configuration Management.

Reference Books:

1. Software Engineering: An Engineering Approach, by J.F.Peters and W. Pedrycz, Publisher: John Wiley and Sons
2. Software Engineering: A Practitioner's Approach by Roger Pressman, Publisher: McGraw-Hill
3. Fundamentals of Software Engineering by Ghezzi, Jayazeri, and Mandrioli, Publisher: Prentice-Hall
4. Software Engineering Fundamentals by Ali Behforooz, and Frederick J.Hudson, Publisher: Oxford University Press

Semester – V
Course Code : BCA/506/INT-3
Course Title : Major Project & Viva-Voce (4 Credit)
Course Objectives:

- To apply the software engineering principles on a real software project
- Develop a software product using the agile methodology.

Learning Outcomes:

After the completion of this course, the students will be able to:

- In a specialization domain of his / her choice, student manager will be able to choose an appropriate topic for study and will be able to clearly formulate& state a research problem.
- For a selected research topic, student manager will be able to compile the relevant literature and frame hypotheses for research as applicable.
- For a selected research topic, student manager will be able to plan a research design including the sampling, observational, statistical and operational designs if any.
- For a selected research topic, student manager will be able to compile relevant data, interpret & analyse it and test the hypotheses wherever applicable.
- Based on the analysis and interpretation of the data collected, student manager will be able to arrive at logical conclusions and propose suitable recommendations on the research problem.
- Student manager will be able to create a logically coherent project report and will be able to defend his / her work in front of a panel of examiners

Theory: 60 Lectures

PROJECT GUIDLINES

Sl. No.	Topic
I	Message from the Project Coordinator
II	Calendar for the Project
III	Performa for BCA (Hons.) (INT-3) Project Proposal (Project's Title and Guide's Details)
IV	Guidelines for Project Formulation
V	Project Proposal Submission and Approval

I. MESSAGE FROM THE PROJECT CO-ORDINATOR

The BCA programme prepares the students to take up positions as Programmers, Systems Analysts, Systems Designers in the field related to computer science and information technology, and ITES or students may go for higher studies in this area. We had therefore imparted the comprehensive knowledge covering the skills and core areas of computer science courses with equal emphasis on the theory and practice in BCA programme.

The BCA students are encouraged to involve themselves completely on the project work in their final semester. It is advised to students to develop their project for solving problems of software industry or any research organization. Doing this will give more exposure to handle real life problems of project development.

The courses studied by you during your BCA programme provide you the basic background to work on diverse application domains. The theoretical background of various courses provides you the necessary foundation, principles, and practices to develop effective ways to solve computing problems. The hands on experience gained from the practical courses provide you the knowledge to work with various operating systems, programming languages, and software tools.

This project work is kept in BCA program to give you opportunity to develop quality software solution. During the development of the project you should involve in all the stages of the software development life cycle (SDLC) like requirements analysis, systems design, software development/coding, testing and documentation, with an overall emphasis on the development of reliable software systems. The primary emphasis of the project work is to understand and gain the knowledge of the principles of software engineering practices, and develops good understanding of SDLC.

Students should take this project work very seriously. INT-3 project should be taken as an opportunity to develop software, which gives exposure to SDLC. Topics selected, should be complex and large enough to justify as a BCA (Hons.) project. The project should be genuine and original in nature and should not be copied from anywhere else. If found copied, the project report will be forwarded to the Exam Discipline Committee of the University as an Unfair means case for necessary action. Students should strictly follow and adhere to the INT-3 project guidelines.

I wish you all the success.

II. CALENDAR FOR THE PROJECT

Sl. No.	Topic	Date
1	Submission of a Guide's Bio-Data and project proposal to the Head of the Department of the College	**1st. August to 14st August**

2	Approval of Project	**15 days** after the project proposal is received.
3	Submission of the Project Report (**Two copies**) in bound form to the Head of the Department of the College	**Latest by 30th November**
4	Viva-Voce to be conducted	**Date Fixed by the University**

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

BANKURA UNIVERSITY

BANKURA, WEST BENGAL – 722 155

III. PROFORMA OF BCA PROJECT PROPOSAL (INT-3)

(Project's Title and Guide's Details)

UID: …… Registration No.: …… College Code: ……

1. Name and Address of the student

……

……

……

E-mail: …… Mob. No. ……

2. Title of the Project

……

3. Name and Address of the Guide

……

……

……

E-mail: …… Mob. No. ……

1. Qualification of the Guide
(Attach Bio-Data also)

Ph.D.	M. Tech.	B. Tech.	MCA	Any other

Note:
1. All the above mentioned Degrees must have been awarded in Computer Science/IT only.
2. A Guide should not guide more than 8 students of BCA at any point of time)

2. Industrial / Teaching experience of the Guide (in Years) ……

3. Software Used for this Project:

……

……

Signature of the Student
Date: ……

Signature of the Guide
Date: ……

Important: 1. Attach this Proforma along with Guide's Bio-data and Project Synopsis in the Project Report.

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

2. Not more than four student is permitted to work on a project.

For Office Use Only

☐ **Approved**	☐ **Not approved**

…….

Signature, Designation, Stamp of the Project Proposal Evaluator

Date: …….

Suggestions for reformulating the Project:

Ensure that you include the following while submitting the Project Proposal:

1. **Proforma for Approval of Project Proposal duly filled and signed by both the student and the Project Guide with date.**
2. **Bio-data of the project guide with her/his signature and date.**
3. **Synopsis of the project proposal (12-15 pages).**

A photocopy of the complete Project Proposal (along with Project Proforma, Project Synopsis, Bio-data of the guide) submitted to your Head of the Department, should be retained by the student for future reference.

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

IV. GUIDELINES FOR PROJECT FORMULATION

The project work constitutes a major component in most of the professional programmes and it is to be carried out with due care and should be executed with seriousness by the candidates.

TYPE OF PROJECT

As majority of the students are expected to work out a real life project in some industry/research and development laboratories/educational institutions/software companies, it is suggested that the project is to be chosen which should have some direct relevance in day-to-day activities of the candidates in his/her institution. Students are encouraged to work in the areas listed at the end. However, it is not mandatory for a student to work on a real life project. The student can formulate a project problem with the help of Guide.

PROJECT PROPOSAL (SYNOPSIS)

The project proposal should be prepared in consultation with your guide. The project proposal should clearly state the project objectives and the environment of the proposed project to be undertaken. **The project work should compulsorily include the software development.** The project proposal should contain complete details in the following form:

1. Title of the Project
2. Introduction and Objectives of the Project
3. Project Category (RDBMS/OOPS/Networking/Multimedia/Artificial Intelligence/Expert Systems/Cyber Security/Block Chain Technology etc.)
4. Analysis (DFDs at least up to second level, ER Diagrams/ Class Diagrams/ Database Design etc. as per the project requirements).
5. A complete structure which includes:
    - Number of modules and their description to provide an estimation of the student's effort on the project.
    - Data Structures as per the project requirements for all the modules.
    - Process Logic of each module.
    - Testing process to be used.
    - Reports generation (Mention tentative content of report)
6. Tools / Platform, Hardware and Software Requirement specifications
7. Are you doing this project for any Industry/Client? Mention Yes/No. If Yes, Mention the Name and Address of the Industry or Client
8. Future scope and further enhancement of the project.

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

V. PROJECT PROPOSAL SUBMISSION AND APPROVAL

After finalizing the topic and the selection of the guide, students should submit the Project Proposal Proforma given along with the synopsis and bio-data of the guide. Incomplete project proposals in any respect will be summarily rejected.

COMMUNICATION OF APPROVAL:

Communication regarding the Approval / Non-approval of the project will be sent to you within four weeks after the receipt of the project proposal.

RESUBMISSION OF THE PROJECT PROPOSAL IN CASE OF NON-APPROVAL

In case of non-approval, the suggestions for reformulating the project will be communicated to you. The revised project synopsis along with a new Performa, should be re-submitted along with a copy of the earlier synopsis and non-approval project proposal Performa in the specific time period**.** These guidelines are applicable for earlier batch students also whose project work is pending.

ELIGIBILITY OF PROJECT GUIDE

1. A person having Ph. D./M. Tech. in Computer Science.

OR

2. A person having B. E/B. Tech. (Computer Science), MCA, M. Sc. (Computer Science/IT) with minimum 2 years' experience in Industry / Teaching.

VI. PROJECT REPORT FORMULATION

ITEMS TO BE INCLUDED IN THE PROJECT REPORT

The following items should be included in the Project Report:

1. The project report must contain the following:
   - Introduction
   - Objectives & scope of the Project
   - Theoretical background
   - Definition of problem
   - Tools/Environment Used
   - System planning (PERT Chart)
   - Cost and benefit analysis
   - Analysis Document (This should include SRS in proper structure based on Software Engineering concepts, E-R diagrams/Class diagrams/any related diagrams (if the former are not applicable), Data flow diagrams/other similar diagrams (if the former is not applicable), Data dictionary)
   - Design Document (Modularization details, Data integrity & constraints including database design, Procedural design, User interface design)
   - Program code (Complete code (well indented)/Detailed specification instead of code*, Comments & Description. The program code should always be developed in such a way that it includes complete error handling, passing of parameters as required, placement of procedure/function statements as needed.)

- Testing (Test case designs are to be included separately for Unit testing, Integration testing, System testing; Reports of the outcome of Unit testing, Integration testing, System testing are to be included separately. Also, details of debugging and code improvement are to be included.)
- Input and Output Screens
- Implementation of Security for the Software developed (In case, you have set up a User Name and Password for your software, you should ensure the security of User Name and Password during transmission to server)
- Limitations of the Project
- Future Application of the Project
- Brief background of the organization where the student has developed the Project.
- Data dictionary. This should give a catalogue of the data elements used in the system/subsystem developed.
    - ✓ The following are the details required. Write NA if not applicable:
    - ✓ Data name
    - ✓ Aliases, if any
    - ✓ Length (Size)
    - ✓ Type (Numeric, Alpha, binary etc.)
- List of abbreviations, figures, tables
- Reference
    - Bibliography
    - Website

*Students who have done their project for any organization are permitted to attach detailed algorithm/specification instead of code, in case, the organization doesn't permit them to attach the code. Student needs to attach letter in the project report from the Project Manager of the project in the organization that they are not permitting student to attach the code. In the absence of such letter, the student needs to attach the code compulsorily.

The project report should be hard bound; should consist of a **Contents** page; all pages of report should be numbered; content should be well organized in a meaningful manner; printouts of text & screen layouts should be original and should not be Xeroxed)

2. Original copy of the Approved Project Proposal Proforma, Synopsis and Guide's Bio-data.

3. Certificate of Originality.

4. The Project Report may be about 50 to 80 double spaced A-4 size typed pages (excluding program code). However, 10% variation on either side is permissible.

5. Soft Copy of the Project on CD/DVD

SUBMISSION OF PROJECT REPORT

Only two copy of the project report is to be submitted to the Head of the Department of the College by the date mentioned in the Calendar for the project.

TYPE OF PROJECT

The majority of the students are expected to work on a real-life project preferably in some industry/ Research and Development Laboratories / Educational Institution / Software Company. Students are encouraged to work in the areas listed at the end (Refer page no.15). However, it is ***not mandatory*** for a student to work on a real-life project. The student

can formulate a project problem with the help of her/his Guide and submit the project proposal of the same. If approved, the student can commence working on it and complete it.

PROJECT EVALUATION

The **Project Report** is evaluated for 50 marks and the **Viva-Voce** is for 30 marks. To be declared successful, the student should secure at least 40% marks in both project report evaluation and viva-voce. Students will be duly intimated about the schedule of viva-voce by a letter from the respective College. An unsuccessful student can either submit the same project after following comments on the assessment sheet or s/he can do a different project. Always, ensure that the DSE-04 project guidelines are followed.

Unfair cases of copied versions of the project synopsis and project reports will be sent to Unfair Means Committee of Bankura University for action.

RESUBMISSION OF THE BCA PROJECT IN CASE OF FAILED STUDENTS

If the student is unsuccessful in the project, s/he should „re-do" the whole cycle, right from the submission of the project synopsis. Students are advised to select a new topic for the project and should prepare and submit the project synopsis to the Regional Centre concerned as per the project guidelines. There are no separate slots for the submission of the project synopsis / project reports for the failed students. Respective submissions of the project synopsis and the project reports should be done strictly as per the "Calendar for the BCA project" given in the project guidelines.

ENQUIRIES

Enquiries regarding the Project Report and Viva-Voce should be addressed to the **Head of the Department of the College.**

VII. IMPORTANT POINTS WHILE PREPARING THE PROJECT REPORT

1. The Project Report should be submitted in A-4 size typed in double space. The Project Report should be hard bound.

2. Ensure that it contains the following:
   - Project Proposal Proforma. All the items should be filled. The signatures of both student and Guide should be present.
   - Project Synopsis. Both Guide and student should sign on the Project Synopsis.
   - Guide's Bio data. The Bio-Data should consist of signature of the Guide. Certificate of Originality
   - All signatures should be accompanied by the date of signature.

3. **If any project report is received in the absence of the items listed above, it will be rejected and returned to students for compliance. Also, violation of Project Guidelines may lead to rejection of the Project.**

4. **Two hard bound original copy of the project report is to be submitted to the head of the dept. of the respective college. One copy of the same Project Report is to be retained with the student and the student is supposed to carry his copy while appearing for viva voce. Spiral binding of Project Report is not permitted.**

5. Xerox copy of the project report is not acceptable.

6. Not more than one student is permitted to work on a Project.

7. If the title of the Project differs from the title mentioned in the Project Proposal, the Project Report will be rejected and will be returned back to the student.

VIII. LIST OF BROAD AREAS OF APPLICATION AND RELATED TOOLS

FRONT END / GUI Tools	Visual Basic, Power Builder, X-Windows (X/lib, X/motif, X/Intrinsic), Oracle Developer 2000,VC++, Jbuilder
RDBMS/BACK END	Oracle, Ingres, Sybase, Progress, SQL Plus, Versant, MY SQL, SQL Server, DB2
LANGUAGES	C, C++, Java, VC++, C#
SCRIPTING LANGUAGES	PERL, SHELL Scripts(Unix), TcL/TK
MIDDLE WARE (COMPONENT) TECHNOLOGIES	COM/DCOM, Active-X, EJB, Rational Rose, MSMQ, BEA, Message Q, MTS, CICS
UNIX INTERNALS	Device Drivers, Pipes, RPC, Threads, Sockets
ARCHITECTURAL CONCEPTS	CORBA, TUXEDO
INTERNET TECHNOLOGIES	DHTML, Java script, VB Script, Perl & CGI script, HTML, Java, Active X, RMI, CORBA, SWING, JSP, ASP, XML, EJB, Java Beans, Java Servlets, Visual Age for JAVA, UML, VRML, WML, iPlanet, ATG, BigTalk, CSS, XSL, Oracle ASP server, VB.Net, AWT, J2EE, LDAP, ColdFusion
NETWORKING TECHNOLOGIES	ATM, Frame Relay, TCP/IP, SNMP, GSM, VoIP, PPP, IP-PSTN, SONET/SDH
WIRELESS TECHNOLOGIES	Blue tooth, Wi-Fi, 3G, 4G, ISDN, EDGE
REALTIME OPERATING SYSTEM / EMBEDDED SKILLS	QNX, LINUX, OSEK, DSP, VRTX, RTXC, Nucleus
OPERATING SYSTEMS	WINDOWS 7/8/10, WINDOWS NT, UNIX, LINUX, IRIX, SUN SOLARIS, HP/UX, PSOS, VxWorks, AS400, AIX, DOS
APPLICATIONS	Financial/ Manufacturing/ Multimedia/ Computer Graphics/ Instructional Design/ Database Management System/ Internet/ Intranet/ Computer Networking Communication Software/E-Commerce/ ERP / MRP/ TCP/IP Internals/ Routing protocols/ Socket Programming/ Implementation of Switches & Routers

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

IX. CERTIFICATE OF ORIGINALITY

This is to certify that the project report entitled _____

_____

Submitted to **Bankura University** in partial fulfilment of the requirement for the award of the degree of **BACHELOR OF COMPUTER APPLICATIONS** is an original work carried out by Mr./ Ms._____ UID: _____ and Registration No. _____ under the guidance of Mr./ Ms._____

The matter embodied in this project is a genuine work done by the student and has not been submitted whether to this University or to any other University / Institute for the fulfilment of the requirement of any course of study.

_____

Signature of the Student

Date .....

UID: .....

Reg. No. .....

_____

Signature of the Guide

Date .....

Name .....

Designation .....

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

X. PROJECT TRAINEE LETTER

Date:

Subject: Project Trainee

Sir,

This is to certify that Mr / Ms_____ whose UID_____ and Registration No. _____ is a student of BCA Course Bankura University and has to do a project in his/her Semester-V starting from July-December session. The project is compulsory for BCA course. S/he has to do a project for 3-4 months in Industry/Research Laboratories under the supervision of a guide preferably from the same organization. During his course, the student has gone through / will go through several theoretical papers such as Data Structures, Database Management System, Programming Languages (C, C++, and Java), TCP/IP Programming, Intranet Administration, Computer Networks, and Software Engineering etc. The student also attended / will also attend practical sessions in all courses in which practical sessions were prescribed for various subjects.

Looking forward for your positive response.

_____

Signature & Name of Project Coordinator
with Date and Stamp

Note: This letter may also be signed by Head of the Department / Principal /TIC of the College

Semester - VI
Course Code : BCA/601/MJC-13
Course Title : Programming in Java (4 Credit)
Course Objectives:
The course is designed to provide complete knowledge of Java language. Students will be able to develop logics which will help them to create programs, applications in Java. Also by learning the basic programming constructs they can easily switch over to any other language in future.
Learning Outcomes:
After the completion of this course, the students will be able to develop applications through Java Programming Language.

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Theory: 60 Lectures

Unit-I: Getting Started: Features of Java; Java Magic; Byte Code.

Unit-II: Basics of Java Keywords: Working of Java; Including Comments; Data Types in Java - Primitives Data Types, Abstract / Derived Data Types; Variables in Java; Naming Variables; Using Classes in Java - Standard for Coding; Declaring Methods in Java; Code to Display Test Value; The main Method; Invoking a Method in Java; Saving, Compiling and Executing Java Programs.

Unit-III: Operators and Control Statements: Operators - Arithmetic Operators, Increment and Decrement Operators, Comparison Operators, Logical Operators, Operator Precedence; Control Flow Statements - If-else Statement, Switch Statement, For Loop, While Loop, Do…While Loop, Break Statement, Continue Statement.

Unit-IV: Arrays and Strings: The String Constructors, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, StringBuffer.

Unit-V: Inheritance, Package and Interface: Inheritance - Types of Relationships, What is Inheritance? Why Generalize? Implementing Inheritance in Java, Access Specifiers, The Abstract Class. Packages - Defining a Package, Understanding CLASSPATH; Interface - Defining an Interface, Some Uses of Interfaces, Interfaces versus Abstract Classes.

Unit-VI: Exception Handling: Definition of an Exception; Exception Classes; Common Exceptions; Exception Handling Techniques.

Unit-VII: Streams in Java: Abstract Streams; Stream Classes; Readers and Writers; Random Access Files; Serialization.

Unit-VIII: Applets: What are Applets? The Applet Class; The Applet and HTML; Life Cycle of an Applet; The Graphics Class; Painting the Applet; User Interfaces for Applet; Adding Components to user interface; AWT Controls.

Unit-IX: Event Handling: Components of an Event; Event Classes; Event Listener; Event-Handling; Adapter Classes; Inner Classes; Anonymous Classes.

Unit-I0: JDBC: Database Management; Mechanism for connecting to a back end database; Loading the ODBC driver.

Unit-XI: RMI, CORBA and Java Beans: Remote Method Invocation (RMI) – Introduction, RMI Terminology; Common Object Request Broker Architecture (CORBA) – Introduction, What is Java IDL? Example: The Hello Client-Server; Java Beans – Introduction, the BeanBox, Running the BeanBox.

Unit-XII: JSP and Servlets: Java Server Pages (JSP) – Introduction, What is needed to write JSP based web application? How does JSP look? How to test a JSP? Servlets – Introduction, History of Web Application, Web Architecture, Servlet Life Cycle.

Unit-XIII Swing: Concepts of Swing; Java Foundation Class (JFC); Swing Packages and Classes; Working with Swing-An Example; Swing Components.

Unit-XIV Networking: Networking in Java; URL Objects.

Unit-XV: Case Studies: Real world case studies: Java based stock market, Fintech Application etc.

Reference Books:

1. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.2009.
2. Herbert Schildt , Java 7, The Complete Reference, , 8th Edition, 2009.
3. Cay S. Horstmann, GaryCornell, "Core Java 2 Volume 1 ,9th Edition, Printice Hall.2012
4. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features)", 9th Edition, Printice Hall.2013
5. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.

Practical

1. Write a Java program to print your name.
2. Write a Java program to add two numbers.
3. Write a Java program to change temperature from Celsius to Fahrenheit.
4. Write a Java program to change temperature from Fahrenheit to Celsius.
5. Write a Java program to find area and perimeter of a rectangle.
6. Write a Java program to find area and perimeter of a circle.
7. Write a Java Program to display whether a number is odd or even.
8. Write a Java Program to check if a number is Positive or Negative.
9. Write a Java program to find maximum of three numbers.
10. Write a Java program to swap two numbers.
11. Write a Java program to convert miles to kilometers.
12. Write a Java program to check whether a year is leap year or not.
13. Write a Java program to implement the concept of inheritance.
14. Write a Java program to show method overloading.
15. Write a Java program to show method overriding.
16. Write a Java program to show method hiding.
17. Create a general class ThreeDObject and derive the classes Box, Cube, Cylinder and Cone from it. The class ThreeDObject has methods wholeSurfaceArea ( ) and volume ( ). Override these two methods in each of the derived classes to calculate the volume and whole surface area of each type of three-dimensional objects. The dimensions of the objects are to be taken from the users and passed through the respective constructors of each derived class. Write a main method to test these classes.
18. Write a program to create a class named Vehicle having protected instance variables regnNumber, speed, color, ownerName and a method showData ( ) to show “This is a vehicle class”. Inherit the Vehicle class into subclasses named Bus and Car having individual private instance variables routeNumber in Bus and manufacturerName in Car and both of them having showData ( ) method showing all details of Bus and Car respectively with content of the super class’s showData ( ) method.
19. Write a Java program to explain “multilevel inheritance.”
20. Create a “circle” class & a “point” class. The coordinates of the circle are given and used within the “circle” class as object of the “point” class. Display the area of circle. 2. Create a class called Time, which has three private instance variables – hour, min and sec. It contains a method called add( ) which takes one Time object as parameter and print the added value of the calling Time object and passes Time object. In the main method, declare two Time objects and assign values using constructor and call the add() method.
21. Create a class called Complex, which has three private instance variables –real and imaginary. It contains a method called add( ) which takes one Complex object as parameter and print the added value of the calling Complex object and passes Complex object. In the main method, declare two Complex objects and assign values using constructor and call the add() method.
22. Design an abstract class having two methods. Create Rectangle and Triangle classes by inheriting the shape class and override the above methods to suitably implement for Rectangle and Triangle class.
23. Write a program in Java to illustrate the use of interface in Java.

24. Write a Java program to show the use of all keywords for exception handling.
25. Write a Java program using try and catch to generate NegativeArrayIndex Exception and Arithmetic Exception.
26. Define an exception called “NoMatchFoundException” that is thrown when a string is not equal to “University”. Write a program that uses this exception.
27. Write a Java program for calculating Factorial. Number should be taken through user input (Using Scanner, BufferedReader both).
28. Design a palindrome class that will input a string from console and check whether the string is palindrome or not.
29. Write a Java program to merge two strings.
30. Write a Java program in which total 4 threads should run. Set different priorities to the thread.
31. Create 4 threads with priority 1,3,5,7 respectively. Update a counter in each of the threads for 10 ms. Print the final value of count for each thread.
32. Design a Java applet that will blink “Hello Applet” message in the client area and play a musical sound in the background with a background picture in client area.
33. Design an applet that will display a text as scrolling marquee. The text can be changed by setting different “PARAMS” value.
34. Write a Java Program to Create a Banner Using Applet.
35. Write a Java Program to Display Clock Using Applet.

Semester - VI
Course Code : BCA/602/MJC-14
Course Title : Internet Systems (4 Credit)
Course Objectives:
This course is intended to teach the basics involved in publishing content on the World Wide Web. This includes the ‘language of the Web’ – HTML, the fundamentals of how the Internet and the Web function, a basic understanding of graphic production with a specific stress on creating graphics for the Web, and a general grounding introduction to more advanced topics such as programming and scripting. This will also expose students to the basic tools and applications used in Web publishing.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Analyse a web page and identify its elements and attributes.
- Create web pages using XHTML and Cascading Style Sheets.
- Build dynamic web pages using JavaScript (Client side programming).
- Create XML documents and Schemas.
- Maintaining a Web site
- Register a Web site on search engines.

Theory: 60 Lectures

Unit-I: Introduction to Internet: Evolution of Internet, concept of Intranet and Internet, Applications of Internet, Types of Connectivity such as dial – up, leased, VSAT. etc., Internet Server and Clients module in various Operating Systems, TCP/IP, Introduction to RFC, Addressing in Internet – IP and Domains, major features of IP, IP datagram, major IP services, IP source routing, value of the transport layer, TCP, major features of TCP, passive and active operation, Internet Service Providers.

Unit-II: E-mail and List-servers: E-mail Networks, E-mail protocols(X.400, SMTP, UUCP), Format of an E- mail message , Description of E-mail Headers, E- mail contents and encoding, E-mail routing, List servers, E-mail clients, POP-3, IMAP-4.

Unit-III: File Transfer Protocol: Introduction to FTP, public domain Software, Types of FTP Servers, FTP clients, Common Commands.

Unit-IV: Telnet: Telnet protocol, Server daemon, Telnet clients, Terminal emulation.

Unit-V: Usenet and Internet Relay Chart Introduction to World Wide Web: Evolution of WWW, Basics Features, WWW Browsers, WWW servers, HTTP & URL's.

Unit-VI: WWW Browsers: Basic features, Bookmarks, history. Progress indicators, Personalization of Browsers, Printing displayed pages and forms, Saving Web pages, Netscape Communicators, Internet Explorer, Search and Downloads.

Unit-VII: Web Publishing: Technology Overview, Web site planning, where to host your Web site, Multiple sites on one server, maintaining a Web site, publishing tools.

Unit-VIII: HTML: Document overview, Header elements, Section Headings, Block- oriented elements, Lists, Inline elements, Visual Mark-up, Hypertext links, Uniform Resource Locators,(URL's), Images, Forms, Tables, Special characters.

Unit-IX: Interactivity Tools: CGI, ActiveX, VB Script, Java Script and java.

Unit-X: Multimedia and Graphics: VRML.

Unit-XI: Search Engines: Technology overview, Popular Search Engines, How to register a Web site on search engines.

Unit-XII: Internet Security: Overview of Internet Security threats, Firewalls, Introduction to AAA.

Unit-XIII: E-commerce: Introduction to E-commerce, Payment Methodology, Security aspects, Standard in electronic payment. E-commerce and Banking, E-commerce and Retailing.

Reference Books:

1. Internetworking with TCP/IP – by D. E. Comer, PHI
2. E-Commerce-Paul A. Murphy, TMH

Practical

1. How to assign IP addresses to the PC connected to the Internet?
2. How to remotely access a Computer with IP address?
3. How to assign IP address automatically using DHCP using Cisco Packet Tracer?
4. Configure Routing Information Protocol using Cisco Packet Tracer.
5. Implement distance Vector Routing [RIP] in any Programming language?
6. How to assign multiple IP address in one LAN Card?
7. How to Configure & Change IP address and DNS?
8. How to Configure Switch/Hub using Cisco Packet Tracer?
9. How to Create Local server on your PC Demonstrate?
10. Create a Web Page to Implement frame tag & set image as background.
11. Create a Registration Form using HTML.
12. Create a Web Page to apply anchor tag for linking Web Pages.
13. Publish a Website on Internet by Hosting Site.
14. Install Web server and publish a website on intranet.
15. Create a Web Page to Implement Table Tag & insert image as a button.
16. How to setup FTP Server using Cisco Packet Tracer?
17. How to Use FTP Command to Copy Files to/from Server?
18. How to differentiate Router and Bridge using Cisco Packet Tracer?
19. How to Connect remote Computer using telnet Command?
20. How to Send email from one PC to other using DHCP in Cisco Packet Tracer?
21. How to Configure http / Web Server using Cisco Packet Tracer?

22. How to do Web Programming using HTML in Cisco Packet Tracer?
23. How to design Calculator using VBScript?
24. How to build Calculator using java script?
25. Socket Programming- TCP Client/Server Socket
26. Socket Programming- UDP Client/Server Socket.
27. Write a Program to implement Caesar Cipher
28. Write a program to implement Affine Cipher.
29. How to make Your own Proxy Server?
30. Shows VPN Configuration Using Cisco Packet Tracer.

Semester - VI
Course Code : BCA/603/MJC-15
Course Title : .NET Technology (4 Credit)
Course Objectives:

- To explore .NET technologies for designing and developing dynamic, interactive and responsive web applications.
- Provide a consistent, object-oriented programming environment whether object code is stored and executed locally, executed locally but web distributed, or executed remotely
- Make the developer experience consistent across widely varying types of apps, such as Windows based apps and Web-based apps.
- Build all communication on industry standards to ensure that code based on .NET Framework integrates with any other code
- Building multi-tier enterprise applications
- Client-side programming: HTTP, CGI, Cookies, JavaScript, HTML, XML

Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the .NET framework.
- Develop a proficiency in the C# and VB.NET programming language.
- Proficiently develop ASP.NET web applications using C# / VB.NET.
- Use ADO.NET for data persistence in a web application.
- To understand the 3-tier software architecture (presentation/client tier, application tier, data tier) and develop multi-tier applications to understand and experiment with the deployment of enterprise applications.
- To develop web applications using a combination of client-side (JavaScript, HTML, XML, WML) and server-side technologies (ASP.NET, ADO.NET).

Theory: 60 Lectures

Unit-I: Introduction to .NET Framework: Introducing VB.NET: New Object Oriented Capabilities- Inheritance- Parameterized Constructors- Overriding- Overloading- Shared Members- Events- Exception Handling-.NET Framework Class Hierarchy-The System Namespace. File I/O: Using the System.IO Hierarchy- Streaming text in and out of Text Files- Object Serialization and Deserialization.

Unit-II: Introduction to ADO.NET: Comparison between ADO & ADO.NET—The difference between Connection Model & Disconnected Model – difference between the DataSet and RecordSet- The Dataset Model. Accessing Data using ADO.NET: dataset-DataAdapterDataRelation. The two Managed Providers: SQL Managed Provider-OLEDB Managed Provider. The ADO.NET Object Model: OleDbConnection /SQLConnectionOleDbCommand/SQLCommand-OleDbDataReader/SQLDataReaderOleDbDataAdapter/SQLDataAdapter-The DataSet. Using the Binding Manager to bind controls to the data - Working with Master-Detail relationship

Unit-III: Differences between ASP and ASP.NET. ASP.NET Web Forms: The code behind Web Form-Separations of content & Business logic-Life Cycle of a Web Form Page-Stages in Web Form Processing

Unit-IV: ASP.NET Server Controls. Web Forms Server Controls Recommendation: Validation Controls-Controls that incorporate logic to validate user inputs like a required field, between ranges, or pattern matching. ASP.NET Data Access: Data Binding Server Controls-Viewing Data Collections in a Grid. ASP.NET Caching Mechanism for caching Dynamic response data. Page Output Caching.

Unit-V: Web Services: Introduction to webservice-Architecture of Web service: Universal Discovery Description and Integration-Web Service Description Language –Accessing webservice using different Clients.

Unit-VI: Developing Windows Forms and Application using VB.net: VB.net language syntax and structure, data types, variables, operators and control statements. Various toolbox controls, Types of projects, Exception handling, Database connectivity, working with menus, dialog box and custom controls.

Reference Books:

1. Chris Ullman, John Kauffman, Beginning ASP.NET 1.1 with VB.NET 2003, Wrox Publication
2. ADO.NET Professional, Wrox Publication
3. Alex Homer, Dave Sussman, Professional ASP.NET 1.1, Wrox Publication
4. .NET Framework, OREILY Publication.
5. Crouch, ASP.NET and VB.NET Web Programming, Pearson Education
6. Deitel and Deitel, Visual Basic.NET How to Program, Pearson Education,2nd edition Greg Buczek, ASP.NET Developer's Guide, Tata McGraw-Hill, 2002.

Practical

1. Using VB.net Develop an application which is similar to notepad using menus.
2. a) Using VB.net Develop an application for facilitating purchasing order.
   b) Using VB.net Develop an application for billing system in coffee shop.
3. a) Using VB.net Develop an application which is similar to login form.
   b) Using VB.net Develop an application for fruits billing.
4. Using VB.net Develop an application using tree view control.
5. a) Using VB.net Develop an application using font dialog control.
   b) Using VB.net Develop an application using color dialog control.
6. Using VB.net Develop an application to display the file selected by the user in a web browser control.
7. Using VB.net Develop an application using the data reader to read from a database.
8. Using ASP.net Design an application for dynamically populating a checkbox list.
9. Using ASP.net Design an application for selecting a single day in the calendar control.
10. Using ASP.net Design an application by using the new scroll bar feature with the panel server control.
11. Using ASP.net Design an application with simple bulleted list control.
12. Using ASP.net Design an application for uploading files using the new file upload control.
13. Using ASP.net Design an application for building a form in the wizard control.
14. Using ASP.net Design an application by using the validation controls.
15. Using ASP.net Design an application using the images, sounds for error notification.
16. Using ASP.net Design an application using the grid view control in an ASP.Net web page.
17. Using ASP.net Design an application for adding an insert command to the sql data source control.
18. Using ASP.net Design a web site using the concept of master pages.
19. Design a web application using ASP.net themes.

20. Using ASP.net develop a project which displays the student information in the relevant fields from the database which already exists.

Semester – VI
Course Code : BCA/604/MJC-16
Course Title : Object Oriented Database Management System (4 Credit)
Course Objectives:
This module aims to provide the students the knowledge of Object Oriented Database Management System.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Gaining knowledge about OODBMS
- Understand and identify the analytical characteristics of OODBMS.

Theory: 60 Lectures

Unit-I: Introduction: A major Change - The Relational Data Model, Object Roles in Databases, Sample uses of Object-oriented Databases, Benefits of Object Orientation.

Unit-II: Object Orientated DBMS: The Object-oriented Data Model - Object-Oriented Data Relationships, Object Identifiers, One-to-Many relationships, Many-to-Many relationships, The IS-A relationship, The Extends relationship, The Whole-Part relationship, Relationship Integrity, ER Diagramming Models for Object-Oriented Relationships - Booch Notation, Unified Modeling Language Integrating Objects into a Relational Database, The Extended Relational Model Approach, The Semantic Database Approach.

Unit-III: The Proposed Object Database Standard: Basic OODBMS Terminology, Understanding Types, External specifiers, Implementations, Primitive Types, Inheritance, Interfaces and Inheritance, Classes and Extensions Objects, Collection Objects, Structured Objects, Creating and Destroying Objects Representing Logical Relationships.

Unit-IV: Semantic Database Models & Systems: The Entity Relationship Model, Relational Model – Tasmania (RMT).

Unit-V: Object Oriented Database Systems: Performance Issues in OODBMS, Application Selection for OODBMS, he Object Oriented Database Paradigm Manifesto, The Mandatory Features, The Optional Features.

Unit-VI: OODBMS Architecture: Introduction, an Overview.

Unit-VII: Introducing object oriented programming: Data and Procedural Abstraction Object Type Overview, Creating Object Types, Object type specification: attributes, methods Declaring and initializing objects, SELF parameter, MAP and ORDER Methods Object type inheritance.

Unit-VIII: Objects in the Database: Introduction to objects in the database, Object Tables - creating Object tables, Inheritance and attribute chaining Object views, Creating Basic Object Views Accessing Column Objects DEREF IS DANDLING TREAT VALUE SYS_TYPEID UTL_REF.

Unit-IX: Large Objects: Introduction to Large Objects Features Comparison, Types of LOBS, LOB Locators, Internal LOBs - BLOB, CLOB, NCLOB, External LOBS - BFILE VARRAY.

Reference Books:

1. *Harrington J.L, (2000) "Object-Oriented Database Design, Clearly Explained", Morgan*
2. *Scott Urman, Ron Hardman, Micheal McLaughlin, Oracle Database 10g PL/SQL Programming, Tata McGraw Hill*
3. *Prabhu S.R, (2003), "Object-Oriented Database Systems", Prentice Hall India*

4. *Feuerstein , Pribyl B., Oracle PL/SQLProgramming, O'Reilly*
5. *McLaughlin , (2008), Oracle Database 11g PL/SQL Programming, Oracle Press*

Semester - VI
Course Code: BCA/605/MN-6A
Course Title: LISP Programming (4 Credit)
Course Objectives:
To understand the programming concepts and constructs of LISP.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand basics of LISP and installation.
- Implement the structure and components of LISP Programme.
- Interpret how to write and implement Functions in programme.
- Create programmes involving arrays and strings.
- Develop programmes related to file operations and error handling.

Theory: 60 Lectures

Unit-I: LISP- Overview, Environment setup, Basic syntax, data types, Macros, variables and constants.

Unit-II: Operators (Arithmetic, comparison, logical and bitwise), Decision structures- cond, if, when and case, Loops- Icop, loop for, do, do-times and do-list

Unit-III: Functions in LISP, Predicates, Number and characters.

Unit-IV: Arrays and strings, sequences, lists, symbols, Vectors, set, Tree and Hash table.

Unit-V: Input/Output, File I/O, Structures, packages, Error Handling and classes (CLOS).

Reference Books:

1. Vindarel. 2021. The Common Lisp Cookbook, Diving into the Programmable programming language. Oreilly Publications

Semester - VI
Course Code: BCA/605/MN-6B
Course Title: Business Intelligence (4 Credit)
Course Objectives:
The student should be made to:

- Be exposed with the basic rudiments of business intelligence system.
- Understand the modeling aspects behind Business Intelligence.
- Understand of the business intelligence life cycle and the techniques used in it.
- Be exposed with different data analysis tools and techniques.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Explain the fundamentals of business intelligence.
- Link data mining with business intelligence.
- Apply various modeling techniques.
- Explain the data analysis and knowledge delivery stages.
- Apply business intelligence methods to various situations.

- Decide on appropriate technique.

Theory: 60 Lectures

Unit-I: Business Intelligence: Effective and timely decisions – Data, information and knowledge – Role of mathematical models – Business intelligence architectures: Cycle of a business intelligence analysis – Enabling factors in business intelligence projects – Development of a business intelligence system – Ethics and business intelligence.

Unit-II: Knowledge Delivery: The business intelligence user types, Standard reports, Interactive Analysis and Ad Hoc Querying, Parameterized Reports and Self-Service Reporting, dimensional analysis, Alerts/Notifications, Visualization: Charts, Graphs, Widgets, Scorecards and Dashboards, Geographic Visualization, Integrated Analytics, Considerations: Optimizing the Presentation for the Right Message.

Unit-III: Efficiency: Efficiency measures – The CCR model: Definition of target objectives- Peer groups – Identification of good operating practices; cross efficiency analysis – virtual inputs and outputs – Other models. Pattern matching – cluster analysis, outlier analysis.

Unit-IV: Business Intelligence Applications: Marketing models – Logistic and Production models – Case studies.

Unit-V: Future of Business Intelligence: Future of business intelligence – Emerging Technologies, Machine Learning, Predicting the Future, BI Search & Text Analytics – Advanced Visualization – Rich Report, Future beyond Technology.

Reference Books:

1. Efraim Turban, Ramesh Sharda, Dursun Delen, “Decision Support and Business Intelligence Systems”, 9 th Edition, Pearson 2013.
2. Larissa T. Moss, S. Atre, “Business Intelligence Roadmap: The Complete Project Lifecycle of Decision Making”, Addison Wesley, 2003.
3. David Loshin Morgan, Kaufman, “Business Intelligence: The Savvy Manager‟s Guide”, Second Edition, 2012.
4. Cindi Howson, “Successful Business Intelligence: Secrets to Making BI a Killer App”, McGraw- Hill, 2007.

Semester - VI
Course Code: BCA/605/MN-6C
Course Title: Cryptography and Web Security (4 Credit)
Course Objectives:

- To understand basics of Cryptography and Network Security.
- To be able to secure a message over insecure channel by various means.
- To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
- To understand various protocols for network security to protect against the threats in the networks.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Provide security of the data over the network.
- Do research in the emerging areas of cryptography and network security.
- Implement various networking protocols.
- Protect any network from the threats in the world.

Theory: 60 Lectures

Unit-I: Introduction: Security trends – Legal, Ethical and Professional Aspects of Security, Need for Security at Multiple levels, Security Policies – Model of network security – Security attacks, services and mechanisms – OSI security architecture – Classical encryption techniques: substitution techniques, transposition techniques, steganography- Foundations of

modern cryptography: perfect security – information theory – product cryptosystem – cryptanalysis.

Unit-II: Symmetric Key Cryptography: MATHEMATICS OF SYMMETRIC KEY CRYPTOGRAPHY: Algebraic structures – Modular arithmetic-Euclid's algorithm- Congruence and matrices -Groups, Rings, Fields- Finite fields- SYMMETRIC KEY CIPHERS: SDES – Block cipher Principles of DES – Strength of DES – Differential and linear cryptanalysis – Block cipher design principles – Block cipher mode of operation – Evaluation criteria for AES – Advanced Encryption Standard – RC4 – Key distribution.

Unit-III: Public Key Cryptography: MATHEMATICS OF ASYMMETRIC KEY CRYPTOGRAPHY: Primes – Primality Testing – Factorization – Euler's totient function, Fermat's and Euler's Theorem – Chinese Remainder Theorem – Exponentiation and logarithm – ASYMMETRIC KEY CIPHERS: RSA cryptosystem – Key distribution – Key management – Diffie Hellman key exchange -ElGamal cryptosystem – Elliptic curve arithmetic-Elliptic curve cryptography.

Unit-IV: IP Security and Key Management: IP Security: Architecture - Authentication header - Encapsulating security payloads - combining security associations - key management.

Unit-V: Security Practice and System Security: Electronic Mail security – PGP, S/MIME – IP security – Web Security – SYSTEM SECURITY: Intruders – Malicious software – viruses – Firewalls.

Unit-VI: Web and System Security: Web Security: Secure socket layer and transport layer security - secure electronic transaction (SET) - System Security: Intruders - Viruses and related threads - firewall design principals – trusted systems.

Reference Books:

1. William Stallings: Cryptography and Network Security- Principles And Practice, 5th Edition, Pearson/PHI, 2011
2. Charlie Kaufman, Radia Perlman and Mike Speciner: "Network Security – Private Communication in a Public World", 2nd Edition, Pearson/PHI, 2002.
3. Eric Maiwald: "Fundamentals of Network Security", 1st Edition, Dreamtech Press, 2003.
4. Whitman: "Principles of Information Security", 3rd Edition, Thomson, 2009.
5. Robert Bragg, Mark Rhodes: "Network Security: The complete reference", 1st Edition, TMH, 2004.
6. Buchmann: "Introduction to Cryptography", 2nd Edition, Springer, 2004

Semester - VI
Course Code: BCA/605/MN-6D
Course Title: Software Project Management (4 Credit)
Course Objectives:
The student should be made to:

- To understand the fundamental principles of software project management.
- To have a good knowledge of responsibilities of project manager.
- To be familiar with the different methods and techniques used for project management.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Apply project management concepts and techniques to an IT project.
- Identify issues that could lead to IT project success or failure.
- Explain project management in terms of the software development process.
- Describe the responsibilities of IT project managers.
- Apply project management concepts through working in a group as team leader
- Be an active team member on an IT project.

Theory: 60 Lectures

Unit-I: Introduction: What is project, What is project Management, The role of project Manager, The project Management Profession, Project life cycle.

Unit-II: Technology Context: A system view of project management, Understanding organizations, Stakeholder management, Project phases and the project life cycle, The context of information technology projects.

Unit-III: project schedule: Developing the project schedule, Project management software tools, Developing the project budget, Finalizing the project schedule and budget, Monitoring and controlling the project, The project communications plan, Project metrics, Reporting performance and progress, Information distribution.

Unit-IV: The importance of project risk management: Risk management planning, Common sources of risk on information technology projects, Risk identification, Qualitative risk analysis, Quantitative risk analysis, Risk response planning, Risk monitoring and control, Using software to assist in project risk management.

Unit-V: The importance of project procurement management: Planning purchase and acquisitions, Planning contracting, Requesting seller responses, Selecting sellers, Administering the contract, Closing the contract, Using software to assist in project management, Outsourcing.

Unit-VI: Change management: The nature of change, The change management plan, Dealing with resistance and conflict.

Unit-VII: Leadership & Ethics in Projects: Project leadership, Ethics in projects, Multicultural projects.

Unit-VIII: Project implementation: Introduction to Project implementation, Administrative closure, Project evaluation.

Reference Books:

1. Information Technology Project Management: Kathy Schwalbe Thomson Publication.
2. Information Technology Project Management providing measurable organizational value Jack Marchewka Wiley India.
3. Applied software project management Stellman & Greene SPD.
4. Software Engineering Project Management by Richard Thayer, Edward Yourdon WILEY INDIA.

Semester - VII
Course Code: BCA/701/MJC-17
Course Title: Theory of Computation (4 Credit)
Course Objectives:

The learning objectives of this course are to: introduce students to the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability.

Learning Outcomes:

At the end of the course students will be able to:

- To use basic concepts of formal languages of finite automata techniques
- To Design Finite Automata's for different Regular Expressions and Languages
- To Construct context free grammar for various languages
- To solve various problems of applying normal form techniques, push down automata and Turing Machines
- To participate in GATE, PGECET and other competitive examinations

Theory: 60 Lectures

Unit-I: Concept of Automation: Definition, concept of sequential circuits, state table & state diagram, concept of synchronous, asynchronous machines.

Unit-II: Finite State Machines: Basic definition, mathematical representation, Moore versus Mealy m/c, capability & limitations of FSM, state equivalence & minimization, machine equivalence, incompletely specified machines, merger graph & compatibility graph, information loss less & inverse machines: testing table & testing graph.

Unit-III: Finite Automation: Preliminaries (strings, alphabets & languages, graphs & trees, set & relations), definition, recognition of a language by an automata - idea of grammar, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Regular Expressions, Equivalence of DFAs, NFAs, and Regular, Expressions, Non-regular languages, Pumping lemma.

Unit-IV: Introduction, definition, derivation trees, simplification, Context-Free Grammar (CFG), Parse Trees.

Unit-V: Pushdown Automata: Definition, moves, Instantaneous Descriptions, Deterministic & Non-Deterministic Push Down Automata (PDA), Acceptance by final state & Empty stack, Equivalence of CFGs and PDAs.

Reference Books:

1. Hopcroft JE. and Ullman JD., “Introduction to Automata Theory, Languages & Computation”, Narosa.
2. Lewis H. R. and Papadimitrou C. H., “Elements of the theory of Computation”, P.H.I.
3. Kain, “Theory of Automata & Formal Language”, McGraw Hill.
4. Kohavi ZVI, “Switching & Finite Automata”, 2nd Edn., Tata McGraw Hill.
5. P. Linz, An Introduction to Formal Language and Automata 4th edition Publication Jones Bartlett, 2006

Semester - VII
Course Code : BCA/702/MJC-18
Course Title : Python Programming (4 Credit)

Course Objectives:

The course is designed to provide complete knowledge of Python Programming. Students will be able to develop logics which will help them to create programs, applications through Python Programming. Also by learning the basic programming constructs they can easily switch over to any other language in future.

Learning Outcomes:

After the completion of this course, the students will be able to develop application through Python Programming.

Theory: 60 Lectures

Unit-I: Introduction to Python Scripting: Why Scripting is Useful in Computational Science, Classification of Programming Languages, Productive Pairs of Programming Languages, Gluing Existing Applications, Scripting Yields Shorter Code and Efficiency, Type-Specification (Declaration) of Variables, Flexible Function Interfaces, Interactive Computing, Creating Code at Run Time, Nested Heterogeneous Data Structures, GUI Programming, Mixed Language Programming, When to Choose a Dynamically Typed Language, Why Python? Script or Program? Application of Python, Concept (immutable).

Unit-II: Basic of Python: Python identifiers and reserved words, Lines and indentation, multi-line statements, Comments, Input/output with print and input functions, Command line arguments and processing command line arguments, Standard data types - basic, none, Boolean (true & False)and numbers, Python strings, Data type conversion, Python basic operators (Arithmetic, comparison, assignment, bitwise logical), Python membership operators (in & not in), Python identity operators (is & is not), Operator precedence, Control Statements, Python loops, Iterating by subsequence index, loop control statements (break, continue, pass), Mathematical functions and constants (import math), Random number functions.

Unit-III: Python Strings: Concept, escape characters, String special operations, String formatting operator, Single quotes,

Double quotes, Triple quotes, Raw String, Unicode strings, Built-in String methods, Python Lists – concept, creating and accessing elements, updating & deleting lists-basic list operations, reverse, Indexing, slicing and Matrices, built-in List functions, Functional programming tools - filter(), map(), and reduce(), Using Lists as stacks and Queues, List comprehensions.

Unit-IV: Python Tuples and Sets: Creating & deleting tuples, Accessing values in a tuple, Updating tuples, delete tuple elements, Basic tuple operations, Indexing, slicing and Matrices, built- in tuple functions, Sets - Concept, operations.

Unit-V: Python Dictionary: Concept (mutable), Creating and accessing values in a dictionary, Updating dictionary, delete dictionary elements, Properties of dictionary keys, built-in dictionary functions and methods.

Unit-VI: Functions: Defining a function (def), Calling a function, Function arguments - Pass by value, Keyword Arguments, default arguments, Scope of variable - basic rules, Documentation Strings, Variable Number of Arguments, Call by Reference, Order of arguments (positional, extra & keyword), Anonymous functions, Recursion, Treatment of Input and Output Arguments, Unpacking argument lists, Lambda forms, Function Objects, function ducktyping & polymorphism, Generators (functions and expressions) and iterators, list comprehensions.

Unit-VII: Files and Directories: Creating files, Operations on files (open, close, read, write), File object attributes, file positions, Listing Files in a Directory, Testing File Types, Removing Files and Directories, Copying and Renaming Files, Splitting Pathnames, Creating and Moving to Directories, Traversing Directory Trees, Illustrative programs: word count, copy file,

Unit-VIII: Python packages: Simple programs using the built-in functions of packages matplotlib, numpy, pandas etc. GUI Programming: Tkinter introduction, Tkinter and Python Programming, Tk Widgets, Tkinter examples. Python programming with IDE.

Unit-IX: Python Database Programming: Connecting to a database, Creating Tables, INSERT, UPDATE, DELETE and READ operations, Transaction Control, Disconnecting from a database, Exception Handling in Databases.

Unit-X: Python Exceptions: Exception handling: assert statement, Except clause - with no exceptions and multiple exceptions, Try - finally, raising exceptions, user-defined exceptions.

Reference Books:

1. Introducing Python- Modern Computing in Simple Packages – Bill Lubanovic, O„Reilly Publication
2. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress
3. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries, et al., Pragmatic Bookshelf, 2/E 2014
4. Introduction to Computer Science Using Python- Charles Dierbach, Wiley Publication Learning with Python ", Green Tea Press, 2002

Practical

1. Write a Python program to interchange first and last elements in a list
2. Write a Python program to swap two elements in a list
3. Write a Python | Ways to find length of list
4. Maximum of two numbers in Python
5. Minimum of two numbers in Python
6. Write a Python program to Find the size of a Tuple
7. Write a Python – Maximum and Minimum K elements in Tuple
8. Write a Python – Sum of tuple elements
9. Write a Python – Row-wise element Addition in Tuple Matrix
10. Create a list of tuples from given list having number and its cube in each tuple

11.Write a Python | Sort Write a Python Dictionaries by Key or Value
12.Handling missing keys in Python dictionaries
13.Write a Python dictionary with keys having multiple inputs
14.Write a Python program to find the sum of all items in a dictionary
15.Write a Python program to find the size of a Dictionary
16.Find the size of a Set in Python
17.Iterate over a set in Python
18.Write a Python – Maximum and Minimum in a Set
19.Write a Python – Remove items from Set
20.Write a Python – Check if two lists have atleast one element common
21.How to get list of parameters name from a function in Python?
22.How to Print Multiple Arguments in Python?
23.Python program to find the power of a number using recursion
24.Sorting objects of user defined class in Python
25.Functions that accept variable length key value pair as arguments
26.Write a Python | Sorting string using order defined by another string
27.Write a Python | Find fibonacci series upto n using lambda.
28.Write a Python program to count Even and Odd numbers in a List.
29.Write a Python | Print an Inverted Star Pattern.
30.Write a Python Program to print digit pattern.

Semester - VII
Course Code : BCA/703/MJC-19
Course Title : Digital Image Processing (4 Credit)
Course Objectives:
To introduce the concepts of image processing and basic analytical methods to be used in image processing. To familiarize students with image enhancement and restoration techniques, To explain different image compression techniques. To introduce segmentation and morphological processing techniques.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the need for image transforms different types of image transforms and their properties.
- Develop any image processing application.
- Understand the rapid advances in Machine vision.
- Learn different techniques employed for the enhancement of images.

Theory: 60 Lectures

Unit-I: Digital Image Fundamentals, Sampling and Quantization, Relationship between Pixels. Image Transforms: 2-D FFT, Properties, Walsh Transform, Hadamard Transform, Discrete Cosine Transform, Haar Transform, Slant Transform, Hotelling Transform, Fourier transforms.

Unit-II: Extension to 2-D, DCT, Walsh, Hadamard transforms. Enhancement and segmentation. Point and region dependent techniques. Image encoding: Fidelity criteria. Transform compression. KL, Fourier, DCT, Spatial compression, Run length coding. Huffman and contour coding.

Unit-III: Restoration: Models: Constrained & Unconstrained, Inverse filtering, Least squares filtering, Recursive filtering.

Reference Books:
1. Digital Image Processing & Analysis, Chanda, PHI
2. Fundamentals of Digital Image Processing, Jain, PHI

3. Image Processing-Analysis & Machine Vission, Sonka, VIKAS
4. Digital Image Processing – Rafael C. Gonzalez, Richard E. Woods, 3rd Edition, Pearson, 2008
5. Digital Image Processing using MATLAB – Rafael C. Gonzalez, Richard E Woods and Steven L. Eddings, 2nd Edition, TMH, 2010.

Semester - VII
Course Code : BCA/704/MJC-20
Course Title : Cloud Computing (4 Credit)
Course Objectives:

- To provide students with the fundamentals and essentials of Cloud Computing.
- To provide students a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real life scenarios.
- To enable students exploring some important cloud computing driven commercial systems and applications.
- To expose the students to frontier areas of Cloud Computing and information systems, while providing sufficient foundations to enable further study and research.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.
- Apply the fundamental concepts in datacentres to understand the trade-offs in power, efficiency and cost.
- Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.
- Analyse various cloud programming models and apply them to solve problems on the cloud.

Theory: 60 Lectures

Unit – I: Introduction- Cloud Computing Fundamentals: Cloud Computing definition, Types of cloud, Cloud services: Benefits and challenges of cloud computing, Evolution of Cloud Computing, usage scenarios and Applications, Business models around Cloud – Major Players in Cloud Computing - Issues in Cloud - Eucalyptus - Nimbus - Open Nebula, CloudSim.

Unit – II: Cloud Services and File System: Types of Cloud services: Software as a Service - Platform as a Service – Infrastructure as a Service - Database as a Service - Monitoring as a Service – Communication as services. Service providers- Google App Engine, Amazon EC2, Microsoft Azure, Sales force. Introduction to MapReduce, GFS, HDFS, Hadoop Framework.

Unit – III: Collaborating with Cloud: Collaborating on Calendars, Schedules and Task Management – Collaborating on Event Management, Contact Management, Project Management – Collaborating on Word Processing, Databases – Storing and Sharing Files- Collaborating via Web-Based Communication Tools – Evaluating Web Mail Services – Collaborating via Social Networks – Collaborating via Blogs and Wikis. 185 CS-Engg&Tech-SRM-2013.

Unit – IV: Virtualization For Cloud: Need for Virtualization – Pros and cons of Virtualization – Types of Virtualization – System Vm, Process VM, Virtual Machine monitor – Virtual machine properties - Interpretation and binary translation, HLL VM - Hypervisors – Xen, KVM, VMWare, Virtual Box, Hyper-V.

Unit – V: Security, Standards, and Applications: Security in Clouds: Cloud security challenges – Software as a Service Security, Common Standards: The Open Cloud Consortium – The Distributed Management Task Force – Standards for application Developers – Standards for Messaging – Standards for Security, End user access to cloud computing, Mobile Internet devices and the cloud.

Reference Books:

1. Bloor R., Kanfman M., Halper F. Judith Hurwitz "Cloud Computing" Wiley India Edition,2010
2. John Rittinghouse & James Ransome, "Cloud Computing Implementation Management and Strategy", CRC Press, 2010
3. Antohy T Velte , Cloud Computing : "A Practical Approach", McGraw Hill,2009
4. Michael Miller, Cloud Computing: "Web-Based Applications That Change the Way You Work and Collaborate Online", Que Publishing, August 2008.
5. James E Smith, Ravi Nair, "Virtual Machines", Morgan Kaufmann Publishers, 2006.
6. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010 2.
7. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wile, 2011
8. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012
9. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010
10. Gautam Shroff, Enterprise Cloud Computing Technology Architecture Applications, Adobe Reader ebooks available from eBooks.com,2010
11. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach, McGraw Hills, 2010.
12. Dimitris N. Chorafas, Cloud Computing Strategies, CRC Press, 2010

Semester - VII
Course Code: BCA/705/MN-7A
Course Title: Neural Network (4 Credit)
Course Objectives:
On completion of this course the students will be able to expose themselves towards intelligence systems and knowledge based systems. It also provides knowledge of learning networks.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the difference between biological neuron and artificial neuron.
- Understand the application areas of neural networks.
- Understand building blocks of Neural Networks.
- Develop neural network models.
- Design and develop applications using neural networks.

Theory: 60 Lectures

Unit-I: Introduction to Artificial Neural Networks: Introduction, Artificial Neural Networks, Historical Development of Neural Networks, Biological Neural Networks, Comparison Between them and the Computer, Comparison Between Artificial and Biological Neural Network Basic Building Blocks of Artificial Neural Networks, Artificial Neural Network (ANN) terminologies.

Unit-II: Fundamental Models of Artificial Neural Networks: Introduction, McCulloch - Pitts Neuron Model, Learning Rules, Hebbian Learning Rule Perceptron Learning Rule, Delta Learning Rule (Widrow-Hoff Rule or Least Mean Square(LMS) Rule, Competitive Learning Rule, Out Star Learning, Boltzmann Based Learning, Hebb Net.
Perceptron Networks: Introduction, Single Layer Perceptron, Brief Introduction to Multilayer Perceptron Networks.

Unit-III: Adaline and Madaline Networks: Introduction, Adaline, Madaline. Associative Memory Networks: Introduction, Algorithms for Pattern Association, Hetero Associative Memory Neural Networks, Auto Associative Memory Network, Bi-directional Associative Memory.

Unit-IV: Feedback Networks: Introduction, Discrete Hopfiled Net, Continuous Hopfiled Net, Relation between BAM and

Hopfiled Nets. Feed Forward Networks: Introduction, Back Propagation Network (BPN), Radial Basis Function Network (RBFN).

Unit-V: Self Organizing Feature Map: Introduction, Methods Used for Determining the Winner, Kohonen Self Organizing Feature Maps, Learning Vector Quantization (LVQ), Max Net, Maxican Hat, Hamming Net.
Adaptive Resonance Theory: Introduction, ART Fundamentals, ART 1, ART2

Reference Books:

1. Sivanandam, S Sumathi, S N Deepa; "Introduction to Neural Networks", 2nd ed., TATA McGraw HILL: 2005.
2. Simon Haykin, "Neural networks A comprehensive foundation", 2nd ed., Pearson Education, 2004.
3. B Yegnanarayana, "Artificial neural networks", 1st ed., Prentice Hall of India P Ltd, 2005.
4. Li Min Fu, "Neural networks in Computer intelligence", 1st ed., TMH, 2003

Semester - VII
Course Code: BCA/705/MN-7B
Course Title: R-Programming for Data Science (4 Credit)
Course Objectives:
To expose the student so that the fundamental concepts of R Programming
Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand the basics in R programming in terms of constructs, control statements, string functions.
- Understand the use of R for Big Data analytics.
- Apply R programming for Text processing.
- Appreciate and apply the R programming from a statistical perspective.

Theory: 60 Lectures

Unit-I: Introduction: Introducing to R – R Data Structures – Help Functions in R – Vectors – Scalars – Declarations – Recycling – Common Vector Operations – Using all and any – Vectorized operations – Filtering – Victoriesed if-then else – Vector Element names.

Unit-II: Matrices: Creating matrices – Matrix Operations – Applying Functions to Matrix Rows and Columns – Adding and deleting rows and columns - Vector/Matrix Distinction – Avoiding Dimension Reduction – Higher Dimensional arrays – lists – Creating lists – General list operations – Accessing list components and values – applying functions to lists – recursive lists.

Unit-III: Data Frames: Creating Data Frames – Matrix-like operations in frames – merging Data frames – Applying functions to Data Frames – Factors and Tables – Factors and levels – Common Functions used with factors – Working with tables – Other factors and table related functions – Control statements – Arithmetic and Boolean operators and values – Default Values for arguments – Returning Boolean Values – Functions are objects – Recursion.

Unit-IV: Classes: S3 Classes – S4 Classes – Managing your objects – Input/output – accessing keyboard and monitor – reading and writing files – accessing the internet – String Manipulation – Graphics – Creating Graphs– Customizing Graphs – Saving Graphs to files – Creating Three-Dimensional plots.

Unit-V: Modelling in R: Interfacing R to other languages – Parallel R – Basic Statistics – Linear Model – Generalized Linear models – Non-linear Models – Time Series and Auto-Correlation – Clustering.

Reference Books:

5. Norman Matloff, —The Art of R Programming: A Tour of Statistical Software Design‖, No Starch Press, 2011.

6. Jared P. Lander, —R for Everyone: Advanced Analytics and Graphics‖, Addison-Wesley Data & Analytics Series, 2013.
7. Mark Gardner, —Beginning R – The Statistical Programming Language‖, Wiley, 2013.
8. Robert Knell, —Introductory R: A Beginner's Guide to Data Visualisation, Statistical Analysis and programming in R‖, Amazon Digital South Asia Services Inc, 2013. Richard Cotton(2013). Learning R, O'Reilly Media.
9. Garret Grolemund (2014). Hands-on Programming with R. O'Reilly Media, Inc.
10. Roger D.Peng (2018). R Programming for Data Science. Lean Publishing.

Practical:

List of Programs:

1. R Expressions and Data Structures.
2. Manipulation of vectors and matrix.
3. Operators on Factors in R.
4. Data Frames in R.
5. Lists and Operators.
6. Working with looping statements.
7. Graphs in R.
8. 3D plots in R.

Semester - VII

Course Code: BCA/705/MN-7C

Course Title: Ethical Hacking (4 Credit)

Course Objectives:

The main objectives of the course are to:

- introduce the ethics and legal aspects of ethical hacking.
- thoroughly discuss the penetration process from information gathering to the actual penetration of a system.
- cover in detail the different techniques, methods, and tools used to penetrate an endpoint system.
- make students aware of common web infrastructure and applications attack vectors.
- expose the students to potential countermeasures and how to evade these when penetrating a system.

Learning Outcomes:

After the completion of this course, the students will be able to:

- be aware of the ethics with respect to ethical hacking and its legal implication.
- understand the penetration process, from utilizing the human and physical element of a system, all the way through information gathering to an actual system penetration.
- Be able to perform a series of zero-touch reconnaissance and information gathering process in order to facilitate an effective penetration of a system and/or site.
- Apply strategies and tactics in order to compromise modern operating systems like Windows family of OSs and the Unix family of Oss.
- Be able to utilize tools and skills to perform penetration testing of modern operating systems.
- Be able to perform privilege escalation attack against Unix family of OSs.
- Understand common web infrastructure and applications attack vectors and be able to utilize tools and skills in order to attack web infrastructures and applications.
- Critically evaluate the potential countermeasures to hacking techniques in order to evade attack detection mechanisms.

Theory: 60 Lectures

Unit-I: Introduction to Ethical Disclosure a) Ethics of Ethical Hacking b) Ethical Hacking and the Legal System c) Proper and Ethical Disclosure

Unit-II: Casing the Establishment a) Information Gathering and Reconnaissance b) Scanning c) Identifying Attack Types: Operating System, Application Level, Misconfiguration of Operating System, Services, and Applications d) Gaining and Maintaining Access e) Covering Tracks

Unit-III: Endpoint and Server Hacking a) Hacking Windows b) Hacking Unix c) Cybercrime and Advanced Persistent Threats.

Unit-IV: Web Application Hacking: Explain XSS attacks, Explain different types of XSS attacks, Explain sql injection attacks, Explain cookie stealing, session hijacking, Show data tampering attacks, Phishing Attack, File upload vulnerability.

Unit-V: IOT Hacking: Explain various attack vectors in IOT hacking.

Reference Books:

1. Gray Hat Hacking The Ethical Hackers Handbook, 4th Edition, D. Regalado, S. Harris, A. Harper, C. Eagle, J. Ness, B. Spasojevic, R. Linn, S. Sims, McGrawHill Osborne.
2. Hacking Exposed 7: Network Security Secrets & Solutions, S. Mcclure, J. Scambray, G. Kurtz, McGrawHill Osborne.
3. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, 2nd Edition, D. Stuttard, M. Pinto, John Wiley & Sons.

Semester - VII
Course Code: BCA/705/MN-7D
Course Title: Software Testing (4 Credit)

Course Objectives:

- To study fundamental concepts in software testing.
- To discuss various software testing issues and solutions in software unit test, integration and system testing.
- To expose the advanced software testing topics, such as object-oriented software testing methods

Learning Outcomes:

After the completion of this course, the students will be able to:

- List a range of different software testing techniques and strategies and be able to apply specific(automated) unit testing method to the projects.
- Distinguish characteristics of structural testing methods.
- Demonstrate the integration testing which aims to uncover interaction and compatibility problems as early as possible.
- Discuss about the functional and system testing methods.
- Demonstrate various issues for object oriented testing.

Theory: 60 Lectures

Unit-I: Introduction: Introduction to quality, software quality, fundamentals of software testing, VV model of testing.

Unit-II: Functional Testing: Boundary value Testing, Equivalence class testing, Decision Table based testing, Retrospection.

Unit-III: Structural Testing: Path Testing, Data Flow Testing, Retrospection

Unit-IV: Levels of Testing, Integration Testing, System Testing, Interaction Testing

Unit-V: Object Oriented Testing: Issues, Class Testing, Object Oriented Integration Testing, Object Oriented System Testing

Unit-VI: Testing Process: Planning, Metrics and Reports, Quantitative and Qualitative Analysis, Improvements.

Reference Books:

1. Software Testing Principles, Techniques and Tools, M.G. Limaye, TMH, (Unit- I and VI)
2. Software Testing A Craftman's Approach, Second Edition, Paul C. Jorgensen, CRC Press. (Unit-II to V)
3. Software testing by Yogesh Singh. Cambridge University Press, 2012
4. Introduction to Software Testing, Paul Ammann, Jeff Offutt, Cambridge University Press.
5. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing, Rex Black, Wiley.
6. Software Testing, Second Edition, Ron Patton, SAMS Software Testing, Perry, Wiley India.
7. Software testing by Sandeep Desai, Abhishek Srivastava. (PHI) EEE edition

Practical:

1. Prepare a small project and submit SRS, design, coding and test plan.
2. Study of any one of the testing tools. (e.g win runner, test direct, etc)
3. MANUAL TESTING for the project
    a. Walkthrough
    b. Whitebox Testing
    c. Blackbox Testing
    d. Unit Testing e. Integration Testing
4. Functional Testing
    a. Boundary value Testing
    b. Equivalence class testing
    c. Decision Table based testing
    d. Cause-effect graphs
5. Structural Testing
    a. Path testing
    b. Data-flow testing
6. Regression Testing (use VTEST tool) using automated testing for website.
7. AUTOMATED TESTING for websites
    a. a. Load Testing (use WAPT)
    b. b. Performance Testing (use WAPT)
8. Implement the process Object oriented testing
9. For automated testing, design the test plan and test cases for integration testing on the given case.
10. For automated testing, design the test plan for unit testing.

Semester - VIII
Course Code : BCA/801/MJC-21
Course Title : Distributed Systems (4 Credit)
Theory: 60 Lectures

Course Objectives:

- This course provides an insight into Distributed systems.
- Topics include- Peer to Peer Systems, Transactions and Concurrency control, Security and Distributed shared memory Course

Learning Outcomes:

After the completion of this course, the students will be able to

- Ability to understand Transactions and Concurrency control.
- Ability to understand Security issues.
- Understanding Distributed shared memory.
- Ability to design distributed systems for basic level applications.

Unit-I: Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges.
System Models: Introduction, Architectural Models, Fundamental Models.

Unit-II: Time and Global States: Introduction, Clocks Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States, Distributed Debugging.
Coordination and Agreement: Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication, Consensus and Related Problems.

Unit-III: Inter Process Communication: Introduction, The API for the Internet Protocols, External Data Representation and Marshalling, Client-Server Communication, Group Communication, Case Study: IPC in UNIX.
Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects, Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.

Unit-IV: Distributed File Systems: Introduction, File Service Architecture, Case Study 1: Sun Network File System, Case Study 2: The Andrew File System.
Name Services: Introduction, Name Services and the Domain Name System, Directory Services, Case Study of the Global Name Services.
Distributed Shared Memory: Introduction, Design and Implementation Issues, Sequential Consistency and IVY case study, Release Consistency, Munin Case Study, Other Consistency Models.

Unit-V: Transactions and Concurrency Control: Introduction, Transactions, Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control.
Distributed Transactions: Introduction, Flat and Nested Distributed Transactions, Atomic Commit Protocols, Concurrency Control in Distributed Transactions, Distributed Deadlocks, Transaction Recovery.

Reference Books:

1. 1. Distributed Systems, Concepts and Design, George Coulouris, J Dollimore and Tim Kindberg, Pearson Education, 41" Edition. 2009.
2. Distributed Systems, Principles and Paradigms, Andrew S. Tanenbaum, Maarten Van Steen, 2nd Edition, PHI.
3. Distributed Systems, An Algorithm Approach, Sukumar Ghosh, Chapman&Hall/CRC, Taylor & Fransis Group, 2007.

Semester - VII
Course Code : BCA/802/MJC-22
Course Title : Unix and Shell Programming (4 Credit)
Course Objectives:

- To provide introduction to UNIX Operating System and its File System.
- To gain an understanding of important aspects related to the SHELL and the process.
- To develop the ability to formulate regular expressions and use them for pattern matching.
- To provide a comprehensive introduction to SHELL programming, services and utilities.

Learning Outcomes:
After the completion of this course, the students will be able to:

- Describe the architecture and features of UNIX Operating System and distinguish it from other Operating System.
- Demonstrate UNIX commands for file handling and process control.
- Write Regular expressions for pattern matching and apply them to various filters for a specific task.
- Analyze a given problem and apply requisite facets of SHELL programming in order to devise a SHELL script to solve the problem

Theory: 60 Lectures

Unit-I: Introduction to unix-Brief History-What is Unix-Unix Components-Using Unix-Commands in Unix-Some Basic

Commands-Command Substitution-Giving Multiple Commands.

Unit-II: The File system –The Basics of Files-What's in a File-Directories and File Names-PermissionsINodes-The Directory Hierarchy, File Attributes and Permissions-The File Command knowing the File Type-The Chmod Command Changing File Permissions-The Chown Command Changing the Owner of a File-The Chgrp Command Changing the Group of a File.

Unit-III: Using the Shell-Command Line Structure-Met characters-Creating New Commands-Command Arguments and Parameters-Program Output as Arguments-Shell Variables- -More on I/O Redirection-Looping in Shell Programs.

Unit-IV: Filters-The Grep Family-Other Filters-The Stream Editor Sed-The AWK Pattern Scanning and processing Language-Good Files and Good Filters.

Unit-V: Shell Programming-Shell Variables-The Export Command-The Profile File a Script Run During Starting-The First Shell Script-The read Command-Positional parameters-The $? Variable knowing the exit Status-More about the Set Command-The Exit Command-Branching Control Structures-Loop Control Structures-The Continue and Break Statement-The Expr Command: Performing Integer Arithmetic-Real Arithmetic in Shell Programs-The here Document(<<)-The Sleep Command-Debugging Scripts-The Script Command-The Eval Command-The Exec Command.

Unit-VI: The Process-The Meaning-Parent and Child Processes-Types of Processes-More about Foreground and Background processes-Internal and External Commands-Process Creation-The Trap Command-The Stty Command-The Kill Command-Job Control, Brief introduction to unix socket programming.

Unit-VII: Linux System Administration and Cloud Deployment.

Reference Books:

1. The Unix programming Environment by Brain W. Kernighan & Rob Pike, Pearson.
2. Introduction to Unix Shell Programming by M. G. Venkateshmurthy, Pearson.
3. Unix and shell programming by B.M. Harwani, OXFORD university press

Online References:

1. https://www.geeksforgeeks.org/introduction-linux-shell-shell-scripting/
2. https://www.tutorialspoint.com/unix/shell_scripting.htm
3. https://www.shellscript.sh/first.html

Practical:

1. Write a Shell program to check the given number is even or odd.
2. Write a Shell program to check and display 10 leap years.
3. Write a Shell program to find the area and circumference of a circle.
4. Write a Shell program to check the given number and its reverse are same.
5. Write a Shell program to check the given string is palindrome or not.
6. Write a Shell program to find the sum of odd and even numbers from a set of numbers.
7. Write a Shell program to find the roots of a quadratic equation.
8. Write a Shell program to check the given integer is Armstrong number or not.
9. Write a Shell program to check the given integer is prime or not.
10. Write a Shell program to generate prime numbers between 1 and 50.
11. Write a Shell program to find the sum of square of individual digits of a number.
12. Write a Shell program to find the sum of cube of individual digits of a number.
13. Write a Shell program to execute various UNIX commands using case statements set of numbers.
14. Write a Shell program to count the number of vowels in a line of text.
15. Write a Shell program to display student grades.
16. Write a Shell program to find the smallest number from a set of numbers.

17. Write a Shell program to find the smallest digit from a number.
18. Write a Shell program to find the sum of all numbers between 50 and 100, which are divisible by 3 and not divisible by 5.
19. Write a Shell program to find the sum of digits of a number until a single digit is obtained.
20. Write a Shell program to find the second highest number from a set of numbers.
21. Write a Shell program to find the second largest digit from a number.
22. Write a Shell program to find the sum of odd digits and even digits from a number.
23. Write a Shell program to find the sum of two numbers using function programming.
24. Write a Shell program to find the largest number between two numbers using function.
25. Write a Shell program to find the largest among three numbers.
26. Write a Shell program to find the largest among 'n' different numbers.
27. Write a Shell program to find the largest digit of a number.
28. Write a Shell program to find the sum of 'n' different numbers.
29. Write a Shell program to find the sum of digits of a number.
30. Write a Shell program to print the reverse of a number.
31. Write a Shell program to find the factorial of a number using for loop.
32. Write a Shell program to generate Fibonacci series.
33. Implement TCP Echo client and TCP Echo server (Iterative).
34. Implement TCP Echo client and TCP Echo server (Concurrent).
35. Write to program to
    a. display name of the host
    b. all IP addresses of the host.
    c. Check whether FTP and HTTP services are running on the system.
    d. Display the name of the service running on port number specified by user.

Semester - VIII
Course Code : BCA/803/MJC-23
Course Title : Database Programming with PL/SQL (4 Credit)
Course Objectives:
The course is designed to provide complete knowledge of PL/SQL Programming. Students will be able to develop logics which will help them to create programs, applications through PL/SQL Programming. Also by learning the basic programming constructs they can easily switch over to any other language in future.
Learning Outcomes:
After the completion of this course, the students will be able to develop application through PL/SQL Programming.

Theory: 60 Lectures

Unit-I: Introduction to PL/SQL: History, Benefits, Creating PL/SQL Blocks

Unit-II: Variables and Datatypes: Defining Variables and Datatypes, Variables in PL/SQL, Recognizing PL/SQL Lexical Units, Recognizing Data Types, Scalar Data Types, Writing PL/SQL Executable Statements, Nested Blocks and Variable Scope, Composite Data types: User-Defined Records, Indexing Tables of Records.

Unit-III: SQL in PL/SQL: Review of SQL DML, Retrieving Data in PL/SQL, Manipulating Data in PL/SQL, Transaction Control Statements.

Unit-IV: Program Structures: Conditional Control: IF Statements, Conditional Control: CASE Statements, Iterative Control: Basic Loops, Iterative Control: WHILE and FOR Loops, Iterative Control: Nested Loops.

Unit-V: Cursors and Parameters: Introduction to Explicit Cursors, Explicit Cursor Attributes, Cursor FOR Loops, Cursors with Parameters, Cursors for UPDATE, Multiple Cursors.

Unit-VI: Exception Handling: Handling Exceptions, Trapping Oracle Server Exceptions, Trapping User-Defined Exceptions, Recognizing the Scope of Exceptions.

Unit-VII: Procedures: Creating Procedures, Parameters in Procedures, Passing Parameters.

Unit-VIII: Functions: Creating Functions, Functions in SQL Statements, Review of the Data Dictionary, Managing Procedures and Functions, Review of Object Privileges, Invoker's Rights and Autonomous Transactions.

Unit-IX: Packages: Creating Packages, Managing Package Concepts, Advanced Package Concepts, Persistent State of Package Variables, Oracle-Supplied Packages.

Unit-X: Triggers: Introduction, Creating DML Triggers, Creating DDL and Database Event Triggers, Managing Triggers.

Unit-XI: Dependencies: Introduction to Dependencies, Understanding Remote Dependencies.

Unit-XII: PL/SQL Compiler: PL/SQL Initialization Parameters, Displaying Compiler Warning Messages, Conditional Compilation, Hiding Your Source Code.

Reference Books:

1. Michael McLaughlin, Oracle Database 12c PL/SQL Programming, McGraw Hill.
2. Bayross Ivan, SQL, PL/SQL the Programming Language of Oracle, SCHAND
3. Groff James,SQL The Complete Reference, 3rd Edition, McGraw Hill
4. Feuerstein, Steven (,Oracle PL/SQL Programming Paperback, O'REILLY

Practical

1. Write a PL/SQL block to show an invalid case-insensitive reference to a quoted and without quoted user-defined identifier.
2. Write a PL/SQL block to show a reserved word can be used as a user-define identifier.
3. Write a PL/SQL block to show the result to neglect double quotation marks in reserved word identifier
4. Write a PL/SQL block to show the result to neglect the case sensitivity of a user defined identifier which is also a reserved word.
5. . Write a PL/SQL block to explain single and multiline comments.
6. Write PL/SQL blocks to show the declaration of variables.
7. . Write a PL/SQL block to show a valid case-insensitive reference to a quoted and without quoted user-defined identifier
8. Write a PL/SQL block to show the operator precedence and parentheses in several more complex expressions.
9. Write a PL/SQL block to create a procedure using the "IS [NOT] NULL Operator" and show AND operator returns TRUE if and only if both operands are TRUE.
10. Write a PL/SQL block to create a procedure using the "IS [NOT] NULL Operator" and show NOT operator returns the opposite of its operand, unless the operand is NULL.
11. Write a PL/SQL block to describe the usage of LIKE operator including wildcard characters and escape character.
12. Write a PL/SQL block to check a number is palindrome or not.
13. Write a PL/SQL block to check a number is Armstrong or not.
14. Write a PL/SQL block to check a number is prime or not.
15. Write a PL/SQL block to display factorial of a number.
16. Write a PL/SQL block to display GCD of two numbers.

17. How would you change the SELECT statement in the following block's cursor so that the block can display the sum of salaries in each department?
18. How would you change the SELECT statement in the following block's cursor so that the block can display the sum of salaries in each department?

19. Rewrite the following block to use a cursor parameter. Then rewrite to use a local module, as well as a cursor parameter.

20. In each of the following PL/SQL blocks, a VALUE_ERROR exception is raised (usually by an attempt to place too large a value into a local variable). Identify which exception handler (if any -- the exception could also go unhandled) will handle the exception by writing down the message that will be displayed by the call to PUT_LINE in the exception handler.

Semester - VIII
Course Code : BCA/804/MJC-24
Course Title : Web Technologies (4 Credit)
Course Objectives:
To comprehend the basics of the internet and web terminologies. To introduce scripting language concepts for developing client-side applications. To practice server-side programming features – PHP, JSP. To know the usefulness of web services.
Learning Outcomes:
After the completion of this course, the students will be able to:

1. Develop a dynamic webpage
2. Connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
3. Write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
4. Write a server side java application called JSP to catch form data sent from client and store it on database

Theory: 60 Lectures

Unit-I: Basics of Internet and Web: The basics of Internet, World Wide Web, Web page, Home page, Web site, Static, Dynamic and Active web page, Overview of Protocols – Simple Mail Transfer Protocol, Gopher, Telnet, Emails, TFTP, Simple Network Management Protocol, Hyper Text Transfer Protocol, Client server computing concepts, URL, MIME, HTTP, Web Programmers Toolbox

Unit-II: Web Client and Web Sever: Web Browser, Browsers e.g., Netscape navigator, Internet Explorer, Mozilla Firefox, ClientSide Scripting Languages- VB Script and Java Script, Active X control and Plug-ins; Web Server Architecture, Image maps, CGI, API web database connectivity-DBC,ODBC.

Unit-III: Web Servers and Servlets: Introduction to Servlets, Lifecycle of a Servlet, JSDK, Deploying Servlet, The Servlet API, The javax. Servlet Package, Reading Servlet parameters, Reading Initialization parameters. The javax.servlet HTTP package, Handling Http Request & Responses, Cookies and Session Tracking.

Unit-IV: Database Access: Database Programming using JDBC, JDBC drivers, Studying Javax.sql.* package, Connecting to database in PHP, Execute Simple Queries, Accessing a Database from a Servlet. Introduction to struts frameworks.

Unit-V: Application Development using JSP: The Anatomy of a JSP Page, JSP Processing. JSP Application Design and JSP Environment, JSP Declarations, Directives, Expressions, Scripting Elements, implicit objects. Java Beans: Introduction to Beans, Deploying java Beans in a JSP page.

Reference Books:

1. Burdman, "Collaborative Web Development", Addison Wesley.
2. Sharma &Sharma, "Developing E-Commerce Sites", Addison Wesley
3. Ivan Bayross, "Web Technologies Part II", BPB Publications.
4. Web Programming, building internet applications, Chris Bates 2nd edition, WILEY Dreamtech (UNITs 1, 2)
5. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES By Marty Hall and Larry Brown Pearson (UNITs 3, 4,5)
6. Programming world wide web-Sebesta, Pearson Education, 2007.
7. Internet and World Wide Web – How to program by Dietel and Nieto PHI/ Pearson EducationAsia.
8. An Introduction to WEB Design and Programming –Wang-Thomson
9. PHP: The Complete Reference Steven Holzner TataMcGraw-Hill.

Semester - VIII
Course Code: BCA/805/MN-8A
Course Title: Introduction to Natural Language Processing (NLP) (4 Credit)
Course Objectives:
This course introduces the theory and methodology of natural language understanding and generation. Topics include stemming, lemmatization, parts of speech tagging, parsing, and machine translation. Employing specialized libraries, students develop applications for topic modeling, sentiment analysis, and text summarization.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Extract information from text automatically using concepts and methods from natural language processing (NLP) including stemming, n-grams, POS tagging, and parsing.
- Develop speech-based applications that use speech analysis (phonetics, speech recognition, and synthesis).
- Analyze the syntax, semantics, and pragmatics of a statement written in a natural language.
- Develop a conversational agent that uses natural language understanding and generation.
- Apply machine learning algorithms to natural language processing.
- Write scripts and applications in Python to carry out natural language processing using libraries such as NLTK, Gensim, and spaCY.
- Design NLP-based AI systems for question answering, text summarization, and machine translation.
- Evaluate the performance of NLP tools and systems.

Theory: 60 Lectures

Unit-I: Introduction to Natural Language Processing: What is-A Natural Language, Natural Language Processing, Language Syntax and Structure, Applications of Natural Language Processing.

Unit-II: Python Programming Review: Intensive review of Python programming, Introduction to Lambda Functions, Using Google Colaboratory.

Unit-III: Python for NLP: Working with Text Data, Introduction Text Processing and Analysis.

Unit-IV: Intro to Text Preprocessing and Wrangling: Text cleaning and Tokenization, Removal of special characters, Case conversion, correcting spellings, Removal of stop words, Stemming, Lemmatization, Introduction to SpaCy.

Unit-V: Text Syntax and Structure: Parts of Speech (POS) Tagging, Shallow Parsing/ Chunking, Dependency and Constituency parsing.

Unit-VI: Feature Engineering: Bag of Words model, Bag of N-Grams model, TF-IDF model, Document Similarity.

Unit-VII: Classification of Text: What is text Classification, Automated Text Classification, Classification Models

Unit-VIII: Summarization and Topic Models: Summarization and Information Extraction, Topic Modeling-Gensim and Scikit-Learn, Automated Document Summarization.

Unit-IX: Text Similarity: Essential concepts of text Similarity, Analyzing Term Similarity, Analyzing Document Similarity

Unit-X: Clustering: Cluster analysis, Clustering Applications.

Unit-XI: Semantic Analysis: Introduction to Semantic Analysis, WordNet, NER Tagger, Analysis of Semantic Representations.

Reference Books:

1. Introduction To Natural Language Processing - A Practical Guide For Beginners, Sakil Ansari
2. Natural Language Processing for GTU 18 Course (VII - CE/CSE - 3170723) - Professional Elective - VI (Paperback, Pranjali Deshpande, Soudamini Patil)
3. Daniel Jurafsky, James H. Martin: "Speech and Language Processing", 2/E, Prentice Hall, 2008.
4. James Allen, "Natural Language Understanding", 2/E, Addison-Wesley, 1994
5. Christopher D. Manning, Hinrich Schutze: "Foundations of Statistical Natural Language Processing", MIT Press, 1999
6. Steven Bird, Natural Language Processing with Python, 1st Edition, O'Reilly, 2009.
7. Jacob Perkins, Python Text Processing with NLTK 2.0 Cookbook, Packt Publishing, 2010.

Semester - VIII
Course Code: BCA/805/MN-8B
Course Title: Big Data Analytics and Visualization (4 Credit)
Course Objectives:
To enable students to have skills that will help them to solve complex real-world problems for decision support.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Build and maintain reliable, scalable, distributed systems with Apache Hadoop.
- Understand Spark framework and explore various ML tools for data processing.
- Apply HIVEQL, PIG techniques to solve big data queries.
- Understand conventional SQL query language and NoSQL.
- Design, build and query MongoDB.
- Visualize big data to perform decision making in real world problems.

Theory: 60 Lectures

Unit-I: Introduction to Big Data: Distributed file system– Big data and its importance, 3Vs of Data Volume, Velocity and Variety, Data sets, Data analysis, Data analytics, Business intelligence, KPI, Big data characteristics, Different types of data, Drivers for big data adoption. Big Data Analysis Techniques: Quantitative analysis, Qualitative analysis, Data mining, Statistical analysis, Machine learning, Semantic analysis, Visual analysis, Case studies.

Unit-II: Hadoop Architecture: Overview of Distributed Database Systems, Hadoop eco-system, Hadoop core components, Hadoop distributions, Developing enterprise applications with Hadoop.

Unit-III: Storing Data in Hadoop: Moving data in and out of Hadoop, HDFS architecture, HDFS files, Hadoop specific file types, HDFS federation and high availability, working with HDFS Commands, Fundamentals of HBASE, Zookeeper concepts and methods to build applications with Zookeeper.

Unit-IV: Introduction to SPARK: Introduction to Data Analysis with Spark, Downloading Spark and Getting Started, Programming with RDDs, Machine Learning with MLlib.

Unit-V: HIVE, HIVQL and PIG: HIVE: Architecture and installation, Comparison with traditional database, HIVQL querying data, Sorting and aggregating, Joins & sub queries, HIVE Vs PIG, PIG: Architecture and installation, Execution Mechanisms, load/store operator, Pig scripts.

Unit-VI: NoSQL and Mongo DB: Introduction, Types of NoSQL databases, Advantages of NoSQL, Use of NoSQL in industry, SQL VS NoSQL, MongoDB: MongoDB Support for dynamic queries, Replications, Sharding, Create Database and Drop Database, Collections and Documents, MongoDB Query Language.

Unit-VII: Data Visualization: Bar Charts, Histograms, Pie Charts, Scatter Plots, Line Plots, Regression.

Reference Books:

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, "Professional Hadoop Solutions",Wiley, ISBN: 9788126551071, 2015.
2. Thomas Erl ,"Big Data Fundamentals-Concepts, Drivers and Techniques", Pearson publication,2016
3. Kyle Banker,Piter Bakkum , Shaun Verch, "MongoDB in Action", Second Edition, Dream tech Press
4. Tom White, "HADOOP: The definitive Guide", O Reilly 2012
5. Alain F. Zuur, Elena N. Ieno, Erik H.W.G. Meesters, "Beginner's Guide to R", Springer 2009

Semester - VIII
Course Code: BCA/805/MN-8B
Course Title: Big Data Analytics and Visualization (4 Credit)
Course Objectives:
To enable students to have skills that will help them to solve complex real-world problems for decision support.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Build and maintain reliable, scalable, distributed systems with Apache Hadoop.
- Understand Spark framework and explore various ML tools for data processing.
- Apply HIVEQL, PIG techniques to solve big data queries.
- Understand conventional SQL query language and NoSQL.
- Design, build and query MongoDB.
- Visualize big data to perform decision making in real world problems.

Theory: 60 Lectures

Unit-I: Introduction to Big Data: Distributed file system– Big data and its importance, 3Vs of Data Volume, Velocity and Variety, Data sets, Data analysis, Data analytics, Business intelligence, KPI, Big data characteristics, Different types of data, Drivers for big data adoption. Big Data Analysis Techniques: Quantitative analysis, Qualitative analysis, Data mining, Statistical analysis, Machine learning, Semantic analysis, Visual analysis, Case studies.

Unit-II: Hadoop Architecture: Overview of Distributed Database Systems, Hadoop eco-system, Hadoop core components, Hadoop distributions, Developing enterprise applications with Hadoop.

Unit-III: Storing Data in Hadoop: Moving data in and out of Hadoop, HDFS architecture, HDFS files, Hadoop specific file types, HDFS federation and high availability, working with HDFS Commands, Fundamentals of HBASE, Zookeeper concepts and methods to build applications with Zookeeper.

Unit-IV: Introduction to SPARK: Introduction to Data Analysis with Spark, Downloading Spark and Getting Started,

Programming with RDDs, Machine Learning with MLlib.

Unit-V: HIVE, HIVQL and PIG: HIVE: Architecture and installation, Comparison with traditional database, HIVQL querying data, Sorting and aggregating, Joins & sub queries, HIVE Vs PIG, PIG: Architecture and installation, Execution Mechanisms, load/store operator, Pig scripts.

Unit-VI: NoSQL and Mongo DB: Introduction, Types of NoSQL databases, Advantages of NoSQL, Use of NoSQL in industry, SQL VS NoSQL, MongoDB: MongoDB Support for dynamic queries, Replications, Sharding, Create Database and Drop Database, Collections and Documents, MongoDB Query Language.

Unit-VII: Data Visualization: Bar Charts, Histograms, Pie Charts, Scatter Plots, Line Plots, Regression.

Reference Books:

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, "Professional Hadoop Solutions",Wiley, ISBN: 9788126551071, 2015.
2. Thomas Erl ,"Big Data Fundamentals-Concepts, Drivers and Techniques", Pearson publication,2016
3. Kyle Banker,Piter Bakkum , Shaun Verch, "MongoDB in Action", Second Edition, Dream tech Press
4. Tom White, "HADOOP: The definitive Guide", O Reilly 2012
5. Alain F. Zuur, Elena N. Ieno, Erik H.W.G. Meesters, "Beginner's Guide to R", Springer 2009

Semester - VIII
Course Code: BCA/805/MN-8D
Course Title: Software Quality Assurance (4 Credit)
Course Objectives:
To introduce an integrated approach to software development incorporating quality management methodologies.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Use Software quality models
- Use Quality measurement and metrics
- Use Quality plan, implementation and documentation
- Use Quality tools including CASE tools
- Use Quality control and reliability of quality process
- Use Quality management system models
- Use Complexity metrics and Customer Satisfaction
- Maintain International quality standards – ISO, CMM

Theory: 60 Lectures

Unit-I: Introduction to Software Quality: Software Quality – Hierarchical models of Boehm and McCall – Quality measurement – Metrics measurement and analysis – Gilb's approach – GQM Model.

Unit-II: Software Quality Assurance: Quality tasks – SQA plan – Teams – Characteristics – Implementation – Documentation – Reviews and Audits.

Unit-III: Quality Control and Reliability: Tools for Quality – Ishikawa's basic tools – CASE tools – Defect prevention and removal – Reliability models – Rayleigh model – Reliability growth models for quality assessment.

Unit-IV: Quality Management System: Elements of QMS – Rayleigh model framework – Reliability Growth models for QMS – Complexity metrics and models – Customer satisfaction analysis.

Unit-V: Quality Standards: Need for standards – ISO 9000 Series – ISO 9000-3 for software development – CMM and CMMI – Six Sigma concepts.

Reference Books:

1. Allan C. Gillies, "Software Quality: Theory and Management", Thomson Learning, 2003. (UI : Ch 1-4 ; UV : Ch 7-8)
2. Stephen H. Kan, "Metrics and Models in Software Quality Engineering", Pearson Education (Singapore) Pte Ltd., 2002. (UI : Ch 3-4; UIII : Ch 5-8 ; UIV : Ch 9-11)
3. Mordechai Ben – Menachem and Garry S.Marliss, "Software Quality", Thomson Asia Pte Ltd, 2003.
4. Mary Beth Chrissis, Mike Konrad and Sandy Shrum, "CMMI", Pearson Education (Singapore) Pte Ltd, 2003.
5. ISO 9000-3 "Notes for the application of the ISO 9001 Standard to software development".

Course Code: BCA/802/RPD-1
Course Research Methodology and IPR (4 Credit)
Course Objectives:
To impart knowledge on formulation of research problem, research methodology, ethics involved in doing research and importance of IPR protection.
Learning Outcomes:
After the completion of this course, the students will be able to:

- Understand that today's world is controlled by Computer, Information Technology, but tomorrow world will be ruled by ideas, concept, and creativity.
- Understand research problem formulation & Analyze research related information and Follow research ethics.
- Correlate the results of any research article with other published results. Write a review article in the field of engineering.
- Appreciate the importance of IPR and protect their intellectual property. Understand that IPR protection provides an incentive to inventors for further research work and investment in R & D, which leads to creation of new and better products, and in turn brings about, economic growth and social benefits

Theory: 60 Lectures

Unit-I: Research Methodology: Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, Samples, Variables, Population, Difference between Research Method and Methodology.

Unit-II: Results and Analysis: Importance and scientific methodology in recording results, importance of negative results, different ways of recording, industrial requirement, artifacts versus true results, types of analysis (analytical, objective, subjective), outcome as new idea, hypothesis, concept, theory, model etc.

Unit-III: Technical Writing: Effective technical writing, how to write a manuscript/ responses to reviewer's comments, preparation of research article/ research report, Writing a Research Proposal - presentation and assessment by a review committee.

Unit-IV: Intellectual Property Rights (IPR): Nature of Intellectual Property: Patents, Designs, Trade Mark and Copyright. Process of Patenting and Development: technological research, innovation, patenting & development. Procedure for grants of patents, Patenting under PCT.

Unit-V: Patent Rights and New Developments in IPR: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications. New Developments in IPR: Administration of Patent System.

Reference Books:

1. Ranjit Kumar, Research Methodology- A step by step guide for beginners, Pearson Education, Australia, 2005.
2. Ann M. Korner, Guide to Publishing a Scientific paper, Bioscript Press 2004.
3. T. Ramappa, "Intellectual Property Rights Under WTO", S. Chand, 2008
4. Kothari, C. R. Research Methodology - Methods and Techniques, New Age International publishers, New Delhi, 2004.
5. Stuart Melville and Wayne Goddard, "Research methodology: an introduction for science & engineering students', Juta & Company, 1996.
6. Robert P. Merges, Peter S. Menell and Mark A. Lemley, "Intellectual Property in New Technological Age", Aspen Publishers, 2016

Course Code: BCA/803/RPD-2
Course : Research and Publication Ethics (4 Credit)
Course Objectives:
To impart knowledge on awareness about the publication ethics and publication misconducts.
Learning Outcomes:
After the completion of this course, the students will be able to:

- To understand the philosophy of science and ethics, research integrity and publication ethics. To identify research misconduct and predatory publications.
- To understand indexing and citation databases, open access publications, research metrics (citations, h-index, impact Factor, etc.). To understand the usage of plagiarism tools.

Theory: 60 Lectures

Unit-I: Philosophy and Ethics: Introduction to philosophy: definition, nature and scope, concept, branches - Ethics: definition, moral philosophy, nature of moral judgments and reactions.

Unit-II: Scientific Conduct: Ethics with respect to science and research - Intellectual honesty and research integrity - Scientific misconducts: Falsification, Fabrication and Plagiarism (FFP) - Redundant Publications: duplicate and overlapping publications, salami slicing - Selective reporting and misrepresentation of data.

Unit-III: Publication Ethics: Publication ethics: definition, introduction and importance - Best practices/standards setting initiatives and guidelines: COPE, WAME, etc. - Conflicts of interest - Publication misconduct: definition, concept, problems that lead to unethical behaviour and vice versa, types - Violation of publication ethics, authorship and contributor ship - Identification of publication misconduct, complaints and appeals - Predatory publisher and journals.

Unit-IV: Open Access Publishing: Open access publications and initiatives - SHERPA/RoMEO online resource to check publisher copyright & self-archiving policies - Software tool to identify predatory publications developed by SPPU - Journal finger / journal suggestion tools viz. JANE, Elsevier Journal Finder, Springer, Journal Suggester, etc.

Unit-V: Publication Misconduct:
Group Discussion: a) Subject specific ethical issues, FFP, authorship b) Conflicts of interest c) Complaints and appeals: examples and fraud from India and abroad.
Software tools (2 Hrs.): Use of plagiarism software like Turnitin, Urkund and other open source software tools.

Unit-VI: Databases and Research Metrics:
Databases: Indexing databases, Citation databases: Web of Science, Scopus, etc.
Research Metrics: Impact Factor of journal as per Journal Citations Report, SNIP, SJR, IPP, Cite Score - Metrics: h-index, g index, i10 Index, altmetrics.

***Units I,II and II are to be covered via Theory mode and Units IV,V and VI are to be covered via practice mode.**

Reference Books:

1. Nicholas H. Steneck. Introduction to the Responsible Conduct of Research. Office of Research Integrity. 2007. Available at: https://ori.hhs.gov/sites/default/files/rcrintro.pdf
2. The Student's Guide to Research Ethics By Paul Oliver Open University Press, 2003
3. Responsible Conduct of Research By Adil E. Shamoo; David B. Resnik Oxford University Press, 2003
4. Ethics in Science Education, Research and Governance Edited by Kambadur Muralidhar, Amit Ghosh Ashok Kumar Singhvi. Indian National Science Academy, 2019. ISBN: 978-81-939482-1-7.
5. Anderson B.H., Dursaton, and Poole M.: Thesis and assignment writing, Wiley Eastern 1997.
6. Bijorn Gustavii: How to write and illustrate scientific papers? Cambridge University Press.
7. Bordens K.S. and Abbott, B.b.: Research Design and Methods, Mc Graw Hill, 2008.
8. Graziano, A., M., and Raulin, M.,L.: Research Methods – A Process of Inquiry, Sixth Edition, Pearson, 2007.

Course Code: BCA/804/RPD-3

Course : Research Project/Dissertation and Viva-voce (4 Credit)

Course Objectives:

- To learn more in-depth knowledge of the major subject/field of study, including deeper insight into current research and development work.
- To contribute to research and development work.
- To use a holistic view to critically, independently, and creatively identify, formulate and deal with complex issues.
- To plan and use adequate methods to conduct qualified tasks in given frameworks and to evaluate this work.
- To create, analyze and critically evaluate different technical/architectural solutions.
- To critically and systematically integrate knowledge.

Learning Outcomes:

After the completion of this course, the students will be able to:

- Ensure proper research and background knowledge is acquainted
- Feasibility study is conducted on the proposed thesis/project
- Scope of the thesis/project is precise and crystal clear
- Generating and comparing alternative designs to determine best match for the requirements
- Roles & responsibilities of individual student working within the group is clear and accepted
- Able to apply thesis/project resources as per the approved thesis/project plan

Guidelines for preparation of Research Project / Dissertation Report

1. Organization of the Research Project / Dissertation Report:

The Research Project/ Dissertation Report shall be presented in a number of chapters, starting with 'Introduction' and ending with 'Conclusion'. The chapters will have a precise title reflecting the contents of the chapter. A chapter can be subdivided into sections, sub-sections to present the content discretely. Total numbers of chapters may be ranged from1- 5(minimum) and 1-8(maximum).

1.1 Introduction:

The title of Chapter-I shall be 'Introduction'. It shall justify the research problem, define the topic and explain the aim and scope of the proposed research. The significant contribution from the investigation to civil society may also be focused in this chapter.

1.2 Review of Literature:
The title of Chapter-II shall be "Review of Literature". This chapter shall present a critical appraisal of the previous works published in the literature pertaining to the topic of the investigation.

1.3 Report on the Present Investigation:
The reporting on the investigation shall be presented in one or more chapters with appropriate title.

1.4 Research Methodology:
Figures and tables should be presented immediately following their first mention in the text. Short tables and figures (say, less than half the writing area of the page) should be presented within the text, while large tables and figures may be presented on separate pages.

1.5 Results and Discussions:
This is the penultimate chapter of the report and shall include a thorough evaluation of the research problem carried out and bring out the contributions from the study. The discussion shall logically lead to meaningful conclusions.

1.6 Conclusions:
This is the last chapter of the report. A brief report of the work carried out shall form the first part of the chapter. Conclusions derived from the logical analysis presented in the "Results and Discussions" chapter shall be presented and clearly enumerated. This chapter should indicate the possibilities/ scope for future work in the concerned field.

NB: In non-empirical research, particularly in the discipline of Humanities and Social Sciences, textual/content analysis may be conceived in several chapters with appropriate title instead of present investigation, results and discussion.

1.7 Appendix:
Detailed information, lengthy derivations, observations etc. are to be presented in the separate appendices, which shall be numbered in Roman Capitals (e.g. "Appendix-I")

1.8 Reference:
This should follow the appendices, (if any), otherwise the Conclusion chapter. The researchers shall follow either the MLA (latest edition), or APA (latest edition) referencing style, or any other style whichever is accepted by the concerned department.

2. Dissertations/ Project Reports Format:

2.1 Paper:
The report shall be in printed form and the size of the paper shall be standard A4; height 297mm, width 210mm.

2.2 Type – Setting, Text Processing and Printing:
The text shall be printed on single side of a page employing laser jet or inkjet printer. The text having been processed using a standard text processor. The standard font should be Time New Roman of 12pts with 1.5 line spacing for English text while Geetanjali of 12pts with 1.5 line spacing and Mangal /Unicode of 12pts with 1.5 spacing for Assamese and Hindi texts respectively.

2.2.1 Page Format :

The Printed sheets shall have the following written area and margins:
Top margin : 15mm
Head height : 3mm
Head separation : 12mm
Bottom margin : 22mm
Footer : 3mm
Foot separation : 10mm
Left margin : 30mm
Right margin : 20mm
Text height : 245mm
Text width : 160mm
When header is not used, the top margin shall be 30mm.

2.2.2 Pagination :

Page numbering in the text of the Dissertation/Report shall be Hindu-Arabic numerals at the right corner of the Page. Page number "1" for the first page of the Introduction chapter should not appear in print; only the second page will bear the number "2". The subsequent chapters shall begin on a fresh page. Pagination for pages before the Introduction chapter shall be in lower case Roman numerals, e.g., "i", " ii" etc.

2.2.3 Header:

When the header style is chosen, the header can have the Chapter number and Section number (e.g., Chapter-II, Section-iii) on even numbered page headers and Chapter title or Section title on the numbered page header.

2.2.4 Paragraph format:

Vertical space between paragraphs shall be about 2.5-line spacing. A paragraph should normally comprise more than one line. A single line of a paragraph shall not be left at the top or bottom of a page.

2.3 Chapter and Section Format:

2.3.1 Chapter:

Each chapter shall begin on a fresh page with an additional top margin of about 75mm. Chapter number (Roman Numerical) and title shall be printed at the center of the line in 6mm font size (18pt) in bold face using both upper and lower case. (See the specimen: 'E')

2.3.2 Section and Sub-sections:

A chapter can be divided into Section and Sub-Sections so as to present different concepts separately. Sections and sub-sections can be numbered using decimal points, e.g. II. ii for the second section in Chapter-II and II. iii. 4 for the fourth Sub-section in third section of Chapter-II.

2.3.3 Table/ Figure Format:

Tables and figures should be presented in portrait style. Small size table and figures (less than half of writing area of a page) should be incorporated within the text, while larger ones may be presented on separate pages. Tables and figures shall be numbered chapter-wise. For example, the second figure in Chapter-IV will bear the number Figure IV.2 or Fig. IV.2.

3.0 Auxiliary Format:

3.1 Binding:

The final hard bound copies to be submitted after the viva-voce examination will be accepted during the submission of Dissertation/Project Report with black colour for P.G and brown colour for U.G course respectively.

3.2 Front Covers:

The front covers shall contain the following details:

- Full title of Dissertation/ Project Report in 6mm 22point's size font properly centered and positioned at the top.
- Full name of the candidate in 4.5mm 15 point's size font properly centered at the middle of the page.
- A 50mm die replica of the institute emblem followed by the name of the supervisor, name of the department, name of the institute and the year of submission, each in a separate line and properly centered and located at the bottom of page

3.2.1 Lettering:

All lettering shall be embossed in gold.

3.2.2 Bound back:

The degree, the name of the candidate and the year of submission shall also be embossed on the bound (side) in gold.

3.3 Blank Sheets:

In addition to the white sheets (binding requirement) two white sheets shall be put at the beginning and the end of the Dissertation/ Project Report.

3.4 Title Sheet:

This shall be the first printed page of the report and shall contain the submission statement: The Dissertation/ Project Report submitted in partial fulfilment of requirements of the------ Degree, the name and Roll No. of the candidate, name(s) of the Supervisor and Co-supervisor(s) (if any), Department, Institute and year of submission.

- Sample copy of the 'Title Sheet' is appended (Specimen 'A')

3.5 Approval Sheet:

This will form the first page of the Dissertation/ Project Report. Sample copy of the 'Internal Approval Sheet' is appended (Specimen-'B').

3.6 Internal Approval Sheet:

This will form the second page. A sample copy of the Approval Sheet is appended (Specimen- C').

3.7 A Declaration of Academic Honesty and Integrity:

A declaration of academic honesty and integrity is required to be included along with every Dissertation/ Project Report after the internal approval sheet. The format of this declaration is given in Specimen-'D' attached.

3.8 Acknowledgements:

A list of acknowledgements is required to be included along with every Dissertation/ Project Report after the

Declaration sheet.

3.9 Contents:

NB: From 3.5 to 3.8, the pagination should be in Roman number with lower case.

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Specimen 'A': Title Sheet

Title

(A Dissertation/Project Report submitted in partial fulfilment of the requirements of BCA degree Honours with Research

by

(Name of the Student)
(Roll No. _____)
Registration No. _____

Supervisor(s):
(Name of Supervisor)
(Name of the Department)
(Name of the College)
Affiliated by
(Name of the University)
(Year)

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Specimen 'B': Approval Sheet

Dissertation Approval for ……

This Dissertation/ Project Report entitled (Title) by (Author Name) is approved for the degree of _____

(Degree details).

Examiners

1. ……
2. ……

Date:
Place:

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Specimen 'C': Internal Approval Sheet

CERTIFICATE

This is to certify that the Dissertation/Project Report entitled "Title of Dissertation/ Project" is a bona-fide work of "Name of student" (Roll No…., Regd. No……..) submitted to the **BANKURA UNIVERSITY** in partial fulfilment of the requirements for the award of **"BCA Honours with Research Degree"**

(Name and Sign)
Supervisor

(Name and Sign)
Co-Supervisor

(Name and Sign)
Head of Department

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Specimen 'D': Declaration

DECLARATION

I, (name), do hereby declare that the dissertation/Project Report entitled "Title" represents my idea in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misinterpreted or fabricated or falsified any idea/ data/ fact/ source in my submission. I understand that any violation of the above will invite disciplinary action by the institute.

……
(Signature)

……
(Name of student and Roll No)

Date :
Place:

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Specimen 'E' : Starting Chapter page Formatting

Chapter – I

Introduction

1.1 (Specimen 'E')
1.2 Formatting Guidelines

Works cited
……

Chapter – II

Title

2.1 ……
2.2 ……

Works cited
……

BANKURA UNIVERSITY

Curriculum and Credit Framework for BCA Programme

(Basic, Honours and Honours with Research)

Specimen 'F' : Standards Style references

References

MLA 9th Edition
Refer to Appendix- I (for quick guide)
APA 7th Edition
Refer to Appendix- II (for quick guide)

========== XXXXX ==========