BANKURA UNIVERSITY

BCA (Hons.)

(As per LOCF (Learning Outcome Based Curriculum Framework)

Under

Choice Based Credit System (CBCS)

SYLLABUS

With Effect from the Session 2022-2023

BANKURA UNIVERSITY
CBCS Syllabus for BCA (Hons.)

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 (Hons.) at Bankura University is designed as per LOCF & as per UGC guidelines.

The main objective of BCA (Hons.) 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 (Hons.) 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 (Hons.) 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 (Hons.) 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 (Hons.) 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 (Hons.) 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 (Hons.) 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

CBCS Syllabus for BCA (Hons.)

Bankura University has started BCA (Hons.) 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 endeavour 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 (Hons.) 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 (Hons.) 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 (Hons.), 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 (Hons.) is aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teaching learning process and in assessment procedures of the learning outcomes. The emphasis in BCA (Hons.) 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 (Hons.) 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 (Hons.) 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

CBCS Syllabus for BCA (Hons.)

Curriculum Planning- Learning Outcomes-based Approach for BCA (Hons.):

BCA (Hons.) 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 (Hons.) Programmes:

The BCA (Hons.) 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 (Hons.) 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 (Hons.) 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 (Hons.) 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 (Hons.):

The BCA (Hons.) 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.

- **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

CBCS Syllabus for BCA (Hons.)

Course Structure of BCA (Hons.)

Semester – I

Sl. No.	Course Code	Course Title	Credit	IA	ESE		Total	L	T	P
1	CC-01	Computer Fundamental and PC Software	6	20	80 / T:50 L:30		100	3	1	4
2	CC-02	Introduction to C Programming	6	20	80 / T:50 L:30		100	3	1	4
3	GE-01	Mathematics-I	6	20	80 / T:80 L:00		100	3	1	-
4	AECC-01	Environmental Studies	4	20	80 / T:80 L:00		100	3	1	-
Total in Semester-I			**22**	**80**	**320**		**400**	**12**	**4**	**8**

Semester - II

Sl. No.	Course Code	Course Title	Credit	IA	ESE		Total	L	T	P
1	CC-03	Digital Logic	6	20	80 / T:50 L:30		100	3	1	4
2	CC-04	Object Oriented Programming using C++	6	20	80 / T:50 L:30		100	3	1	4
3	GE-02	Principals of Accounting and Costing	6	20	80 / T:80 L:00		100	3	1	-
4	AECC-02	English Language and Communication	2	20	80 / T:80 L:00		100	3	1	-
Total in Semester-II			**20**	**80**	**320**		**400**	**12**	**4**	**12**

Semester - III

Sl. No.	Course Code	Course Title	Credit	IA	ESE		Total	L	T	P
1	CC-05	Operating System	6	20	80 / T:50 L:30		100	3	1	4
2	CC-06	Database Management Systems	6	20	80 / T:50 L:30		100	3	1	4
3	CC-07	Data Structure through C++	6	20	80 / T:50 L:30		100	3	1	4
4	GE-03	Mathematics – II	6	20	80 / T:80 L:00		100	3	1	-
5	SEC-01 (A/B/C)	**Choose Any One** • **A.** Android Programming • **B.** PHP Programming • **C.** Digital Image Processing	2	20	80 / T:50 L:30		100	3	1	4
Total in Semester-III			**26**	**100**	**400**		**500**	**15**	**5**	**16**

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester – IV

Sl. No.	Course Code	Course Title	Credit	IA	ESE		Total	L	T	P
1	CC-08	Computer Networks	6	20	80 T:50 L:30		100	3	1	4
2	CC-09	Computer Graphics and Multimedia	6	20	80 T:50 L:30		100	3	1	4
3	CC-10	Theory of Computation	6	20	80 T:80 L:00		100	3	1	-
4	GE-04	Mathematics-III	6	20	80 T:80 L:00		100	3	1	-
5	SEC-02 (A/B/C)	**Choose Any One** • **A.** Python Programming • **B.** Wireless Mobile Communication • **C.** Database Programming with PL/SQL	2	20	80 T:50 L:30		100	3	1	4
Total in Semester-IV			**26**	**100**	**400**		**500**	**15**	**5**	**12**

Semester - V

Sl. No.	Course Code	Course Title	Credit	IA	ESE		Total	L	T	P
1	CC-11	Software Engineering	6	20	80 T:80 L:00		100	3	1	-
2	CC-12	Programming in Java	6	20	80 T:50 L:30		100	3	1	4
3	DSE-01	Introduction to Microprocessor and System Software	6	20	80 T:50 L:30		100	3	1	4
4	DSE-02	Data Warehousing and Data Mining	6	20	80 T:80 L:00		100	3	1	-
Total in Semester-V			**24**	**80**	**320**		**400**	**12**	**4**	**8**

Semester - VI

Sl. No.	Course Code	Course Title	Credit	IA	ESE		Total	L	T	P
1	CC-13	Internet Systems	6	20	80 T:50 L:30		100	3	1	4
2	CC-14	Intelligent Systems	6	20	80 T:50 L:30		100	3	1	4
3	DSE-03	.NET Technology	6	20	80 T:50 L:30		100	3	1	4
4	DSE-04	Major Project & Viva-Voce	6	20	80 T:00 L:80		100	3	1	4
Total in Semester-VI			**24**	**80**	**320**		**400**	**12**	**4**	**16**

Note:

SH = Science Honours, BCA = Bachelor of Computer Application, CC = Core Course, AECC = Ability Enhancement Compulsory Course, SEC = Skill Enhancement Course, GE = Generic Elective, DSE = Discipline Specific Elective, IA = Internal Assessment, ESE= End-Semester Examination, L = Lecture, T = Tutorial, and P = Practical

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)
QUESTION PATTERN

Theory Papers:

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

Full Marks: 80 (Major Project and Viva-Voce)

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

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester - I
Course Code: AECC-01
Course Title: Environmental Studies (4 Credit)
Course Objectives:

- Students will integrate knowledge from multiple disciplines representing physical and life sciences perspectives, political and economic perspectives, and social and cultural perspectives on humans' interactions with their environments;
- Students will contribute to and facilitate interdisciplinary research and problem solving, through independent and collaborative work; and
- Students will use quantitative and qualitative research tools and techniques to analyse, implement, envision, assess, and report sustainability efforts.

Learning Outcomes: After completing the major in Environmental Studies, students will be able to:

- Articulate the interconnected and interdisciplinary nature of environmental studies;
- Demonstrate an integrative approach to environmental issues with a focus on sustainability;
- Use critical thinking, problem-solving, and the methodological approaches of the social sciences, natural sciences, and humanities in environmental problem solving;
- Communicate complex environmental information to both technical and non-technical audiences;
- Understand and evaluate the global scale of environmental problems; and
- Reflect critically on their roles, responsibilities, and identities as citizens, consumers and environmental actors in a complex, interconnected world.

Theory: 60 Lectures

Unit-1: Introduction to Environmental Studies:
- Multidisciplinary nature of environmental studies
- Definition, Nature, Scope and Importance of environmental studies
- Types and Components of environment
- Sustainable development

Unit-2: Ecosystems:
- Concept of Ecology and Eco-system, Structure and Function of an Ecosystem
- Different types of ecosystem; Forest, Desert and Aquatic (Ponds and Oceans) Biomes
- Energy flow in the ecosystem, energy flow models
- Food chains, food weds and ecological pyramids
- Ecological Succession

Unit-3: Natural Resources: Renewable and Non- Renewable Resources:
- Land resources: Land degradation, Landslides, Soil erosion
- Forest resources: Uses, types and importance, deforestation and its effects, Forest bio diversity and tribal population
- Water resources: Distribution of water on Earth; Use and over-exploitation of surface and   ground water; conflicts over water (international & inter-state)
- Energy resources: Renewable and Non-renewable energy sources; Use of alternative energy Sources

Unit-4: Biodiversity and conservation:
- Introduction – Definition: Levels of biological diversity: Genetics, Species and Eco-System

Diversity, Biodiversity hot spots and mega biodiversity countries.
- Threats to biodiversity; Value (services) of biodiversity; man-wildlife conflicts, biological invasions
- Conservation of biodiversity: *In situ* and *Ex situ* conservation of biodiversity; Endangered and endemic species of India

Unit-5: Environmental Pollution:
- Environmental pollution: types, causes, effects and controls; Air, water, soil and noise pollution
- Nuclear hazards and human health risks.
- Solid waste management: Control measures of urban and industrial waste.
- Fireworks Pollution

Unit-6: Environmental Policies & Practices:
- Climate change, global warming, ozone layer depletion, acid rain and its impacts on human communities and agriculture
- Environment Laws: Environment Protection Act, 1986; Air (Prevention & Control of Pollution) Act, 1981; Water (Prevention and control of Pollution) Act, 1972; Wildlife Protection Act, 1972; Forest Conservation Act, 1920, 1988; International agreements: Montreal protocols, 1987 and Kyoto protocols, 1997 and Convention on Biological Diversity (CBD)
- Tribal populations and rights.

Unit 7: Human Communities and the Environment:
- Human population growth: Population Explosion, Impacts on environment, human health and welfare.
- Disaster management: floods, earthquake, cyclones and landslides.
- Environmental movements: Chipko, Silent valley
- Environmental ethics: Role of Indian and other religions and cultures in environmental Conservation
- Environment and human health: Concept of health and diseases (Vector Borne Diseases)
- Human Rights, Value Education, Role of Information Technology in Environment

Unit 8: Field Work (Project Work):
- Visit to an area to document environmental assets: river/ forest/ flora/fauna, etc.
- Visit to a local polluted Site-Urban/Rural/Industrial/Agricultural.
- Study of common plants, insects, birds and basic principles of identification
- Study of simple ecosystems-pond, river etc.

Reference Books:
1. Singh, J. S. Singh, S.P. and Gupta, S.R. 2014.Ecology, Environmental Science and Conservation. S. Chand Publishing, New Delhi.
2. Santra S. C. 2005. Environmental Science, New Central Book Agency (P) Ltd. Kolkata.
3. Singh, S. 1991. Environmental Geography, Prayag Pustak Bhawan, Allahabad.
4. Roy, S. 2003.Environmental Science, Publishing Syndicate, Kolkata
5. Environmental Studies—Prof S.V.S Rana.--Rastogi Publication.
6. Cunningham, W. P. Cooper, T. H. Gorhani, E & Hepworth, M.T.2001, Environmental Encyclopedia. JaicoPubl. House. Mumbai.1196p
7. Erach Bharucha, 2016. Text Book of Environmental Studies for Undergraduate Courses (Second Edition) for UGC. University Press.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester - II
Course Code: AECC-02
Course Title: English Language and Communication (2 Credit)
Course Objectives:

- Ability to be comfortable with English in use while reading or listening.
- Ability to use receptive skills through reading and listening to acquire good exposure to language and literature.
- Ability to write and speak good English in all situations.
- Students should develop style in speech and writing and manipulate the tools of language for effective communication.
- The course should provide exposure to the learners in Good Prose texts and Poems and expose the learners to value based ideas.
- Students should enhance their language skills especially in the areas of grammar and pronunciation.

Learning Outcomes:

- Students can read and understand any text in English listening to the inputs given by the teacher in the classroom.
- Students imbibe the rules of language unconsciously and tune to deduce language structure and usage.
- Students write paragraphs, essays, and letters.
- Students decipher the mechanism of language and use it for success in competitive examinations and job related speaking and writing tasks

Theory: 60 Lectures

Unit-1: Use of language- Human and animals: Basic skills of language Learning, Listening, Speaking, Reading and Writing Definition, purpose and types of communication.

Unit-2: Communication Process: Encoding decoding, message, media-feedback in communication. Effective communication Barriers to Communication. Verbal and non-verbal Communication.

Unit-3: Business Communication: Group Discussion, Seminar, Report and Interview.

Unit-4: Writing Letters: Official and Business Writings- Applications for job- Resume, CV- Report writing.

Unit-5: Speaking and Presentation: Face to face talk advertisement and multimedia. Technical knowledge and skill in modern e-age communication various types of modern communication fax, e-mail, video, Internet, WhatsApp.

Unit-6: Grammar and Grammatical Structures: Vocabulary – Punctuation, Narration – Transformation of sentences – question tags- Correction of errors.

Unit-7: English as a non-phonetic language sounds phonetics- phonetic transcription word stress.

Unit-8: Comprehension: Writing short composition, précis.

Unit-9: Practice and Assignment in the course modules.

Reference Books:
1. Fluency in English - Part II, Oxford University Press, 2006.
2. Business Communication and Report Writing- Sharma, TMH.
3. English for technical Communication – Laxminarayana, Seitech.
4. Business Communication – Kaul, PHI.
5. Communication Skill – Ghanekar, EPH
6. English for the secretary- Yvonne Hoban – TMH
7. English Phonetics –Balashubrahmaniam.

Semester - I
Course Code: CC-01
Course Title: Computer Fundamental and PC Software (6 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 discovers 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
- 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

Theory: 60 Lectures

Unit-I: Introduction: Introduction to computers, Evolution of computer, a brief history, Generation of computers, Classification of computers, The computer system, Application of computers.

Unit-II: Computer System Architecture: Introduction, Central Processing Unit, Communication between various units, Primary Memory (Memory Hierarchy, RAM, ROM), Secondary Storage (Magnetic tape, Magnetic Disk, Optical Disk)

Unit-III: Introduction to Software: Definition of software, Classification of Software, Introduction to Operating System - Definition of OS, Functions of OS, basic concept of different type of OS, Application Software - Definition of Application Software, Types of Application Software, Programming Languages, Machine Language, Assembly Language, High Level Language.

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

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Unit-V: 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 on desktop, closing windows, renaming icons, resizing windows (maximizing and minimizing). Control panel.

Unit-VI: 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-VII: 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, Additional formatting commands and drawing toolbars, Other commands and functions

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

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

Reference Books:
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.
4. 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.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

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 - I
Course Code: CC-02
Course Title: Introduction to C Programming (6 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-1: 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-2: 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-3: 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-4: 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-5: 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-6: 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-7: 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-8: 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.

Reference Books:
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:
5. 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)?
6. Write a C program to find all odd numbers between 150 to 300.
7. Write a C program to find LCM of two (2) number.
8. Write a C program to find all 4-digit number which are divisible by 5, 7 and 3.
9. Develop a program to compute the roots of a quadratic equation by accepting the coefficients. Print appropriate messages?
10. Write a C program to print the following pattern.
    ```
    *
    * *
     * * *
    * * * *
     * * * * *
    ```

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

    * * * * *

    * * * *

    * * *

    * *

    *

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

    * * * * * * *

    * * *   * * *

    * *   * *

    *   *

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

    1

    2 6

    3 7 10

    4 8 11 13

    5 9 12 14 15

14. 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

15. 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.

16. 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!) + …….

17. Design develop and Write c program compute cosine (x) .(hint: 1 - (1/2!)x2 + (1/4!)x4 - (1/6!)x6) using Taylor series.

18. Write a program in C to find the factorial of a number without using recursive C function.

19. Write a c –code to determine whether the given number is prime or not using function.

20. Develop a program to find the reverse of a string and check for palindrome or not Display appropriate messages?

21. 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.

22. Implement using functions to check whether the given number is prime and display appropriate messages (No built-in math function)

23. Develop a program to introduce 2D Array manipulation and implement Matrix multiplication and ensure the rules of multiplication are checked.

24. Write a C program to implement 2D array using pointer.

25. 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

26. Develop a program to sort the given set of N numbers using any sorting algorithm.

27. 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).

28. 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

29. Develop a program using pointers to compute the sum and standard deviation of all elements stored in an array of n real numbers.

30. Write a C program to sort an array of structure (roll, name, total marks) in ascending order of total marks.

31. Implement Recursive functions for binary to Decimal Conversion

32. Write a function to implement string operations such as compare, concatenate, string length. Convince the parameters passing techniques

33. Write a C program to insert five (5) records in a file and display it.

34. 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: CC-03
Course Title: Digital Logic (6 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 analyse combinational logic circuits.
- Be able to design and analyse 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.

Reference Books:
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 generator 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 - II
Course Code: CC-04
Course Title: Object Oriented Programming using C++ (6 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:

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

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

Theory: 60 Lectures

Unit-1: 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-2: Data types, Expression and control statements Iteration statements in C++, Introduction to Arrays, Multidimensional Arrays, Strings and String related Library Functions.

Unit-3: Functions: Introduction to functions, Passing Data to Functions, Scope and Visibility of variables in Functions, Structures in C++.

Unit-4: 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-5: 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-6: 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-7: 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-8: Pointers: Pointers to objects, Virtual Functions, Polymorphism, Abstract classes.

Unit-9: Files and streams in C++: Character and String input and output to files, Command Line Arguments and Printer Output.

Unit-10: 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-11: 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-12: Templates: Function templates and class templates

Unit-13: Standard Template Library: Containers, iterators and application of container classes.

Unit-14: Exception handling: Throwing an exception, catching an exception: The try block, Exception handlers, Termination vs. Resumption, Exception specification, rethrowing an exception, uncaught exceptions, Standard exceptions, Programming with exceptions.

Reference Books:
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. Write a C++ program to print positive number entered by the user If the user enters a negative number, it is skipped.
16. Write a C++ program to show the concept of function overloading to calculate area where same name function differs in number of parameter.
17. Write a C++ program to illustrate the use of constructor member function to initialize and object during its creation.
18. Write a C++ program to understand use of copy constructor.

19.   Write a C++ program to understand the use of constructor overloading and destructor.
20.   Write a C++ program to illustrate how to derive a class from a base class.
21.   Write a C++ program to implement multilevel inheritance.
22.   Write a C++ program to implement hierarchical inheritance
23.   Write a C++ program to illustrate the concept of overriding member function.
24.   Write a C++ program to demonstrate the use of protected members.
25.   Write a C++ program to implement runtime polymorphism using virtual function.

Semester - III
Course Code: CC-05
Course Title: Operating System (6 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-1: 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-2: 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-3: Process Management: Process, Process State, Process Control Block, Process Scheduling, Operation on processes, Co-operating Processes, Threads.

Unit-4: 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-5: 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-6: 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-7: 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-8: 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-9: 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-10: 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-11: 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-12: Security and Protection: Attacks on Security, Computer Worms, Computer Virus, Security Design Principles, Authentication, Protection Mechanism, Encryption, Security in Distributed Environment.

Unit-13: Unix Editors and commands: ed editor, vi editor, Redirections, piping, tees, filters, UNIX utilities: grep, sed, awk, tr etc.

Unit-14: 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-15: Awk programming: Awk arithmetic and variables, Awk built-in variable names and operators, arrays, strings.

Reference Books:
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
8. Sed and Awk, Dale Dougberty and Arnold Robbins, O'REILLY

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.
13. Write a shell Script to find who are currently logged in in the system (display only the name).
14. Write a shell Script to merge contents of two files into a file, now find a particular string in the file.
15. Find the difference between echo $PATH; echo "$PATH"; echo '$PATH'; echo *; echo "*"; echo '*';
16. Write an AWK code to print only certain columns from the input file.
17. Write an AWK code to Print the lines which match the user given pattern
18. Write an AWK code to make Use of NR built-in variables.
19. Write an AWK code to print the first item along with the row number (NR) separated with " – " from each line from a given file.
20. Write an AWK code to return the second column/item from a given file.

Semester - IV
Course Code: CC-06
Course Title: Database Management Systems (6 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-1: 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-2: 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 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-3: 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-4: 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-5: 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-6: SQL: Categories of SQL Commands; Data Definition; Data Manipulation Statements, SELECT - The Basic Form, Sub queries, 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-7: Relational Algebra: Basic Operations, Union (U), Difference (-), Intersection ($\cap$), Cartesian Product (x); Additional Relational Algebraic Operations, Projection (π), Selection (σ), JOIN ( ), Division ($\div$)

Unit-8: Relational Calculus: Tuple Relational Calculus, Semantics of TRC Queries, Examples of TRC Queries; Domain Relational Calculus; Relational ALGEBRA vs Relational CALCULUS

Unit-9: 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-10: Indexing and Hashing: Introduction, Overview, Primary Secondary Multi level, Dense and Space Index.

Unit-11: Distributed Databases: Structure of Distributed Database; Trade-offs in Distributing the Database, Advantages of Data Distribution, Disadvantages of Data Distribution; Design of Distributed Databases, Data Replication, Data Fragmentation

Unit-12: Object Oriented DBMS: Next Generation Data Base System, New Database Application; Object Oriented Database Management System; Features of Object Oriented System; Advantages of Object Oriented Database Management System; Deficiencies of Relational Database Management System; Difference between Relational Database Management System and Object Oriented Database Management System, Alternative Object Oriented Database Strategies

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.

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. 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 - III
Course Code: CC-07
Course Title: Data Structure through C++ (6 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-1: 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; Frequency count, Complexity measures in terms of time and space. Structured approach to programming.

Unit-2: 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-3: 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-4: Abstract Data types Stacks and Queues: Definition of ADT, Stack ADT (array implementation), FIFO queue ADT (array implementation)

Unit-5: 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-6: 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-7: 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).

Reference Books:
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 polynomials.
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: CC-08
Course Title: Computer Networks (6 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, analyse and evaluate a number of datalink, network, and transport layer protocols.
- Program network communication services for client/server and other application layouts.
- Describe, analyse 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, analyse, and evaluate networks and services for homes, data centres, IoT/IoE, LANs and WANs.

Theory: 60 Lectures

Unit-1: 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-2: 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 signalling 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-3: 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-4: 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-5: 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-6: 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-7: 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-8: 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-9: 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-10: 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-11: 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-12: Wireless Networks: Radio Communications, Cellular Radio, Mobile Telephony (GSM & CDMA), Satellite Networks (VSAT), Mobile Adhoc Networks (MANET)

Unit-13: Security and Management: Cryptography, IPsec, SSL/TLS, PGP, secure HTTP, proxy, firewall, VPN; Simple Network Management Protocol (SNMP), Network policies.

Reference Books:
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

8. Practical Computer Network Analysis and Design (The Morgan Kaufmann Series in Networking) by James D. McCabe

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: CC-09
Course Title: Computer Graphics and Multimedia (6 Credit)
Course Objectives:

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

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-1: 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-2: 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-3: Scan conversion Points & lines: Line drawing algorithms; DDA algorithm, Bresenham's line algorithm,

Unit-4: Scan conversion -2: Circle generation algorithm; Ellipse generating algorithm; scan line polygon, fill algorithm, boundary fill algorithm, flood fill algorithm.

Unit-5: 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-6: 2D Viewing: Viewing pipeline, Window to viewport Co-ordinate transformation, clipping operations, point clipping, line clipping, clipping circles, polygons & ellipse.

Unit-7: 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-8: Curves: Curve representation, surfaces, designs, Bezier curves, B-spline curves, end conditions for periodic B-spline   curves, rational B-spline curves.

Unit-9: 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-10: Color & shading models: Light & color model; interpolative shading model; Texture;

Unit-11: Graphics Languages: GKS, PHIGS

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Unit-12: 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 pixels 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.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester - III
Course Code: CC-10
Course Title: Theory of Computation (6 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-1: Concept of Automation: Definition, concept of sequential circuits, state table & state diagram, concept of synchronous, asynchronous machines.

Unit-2: 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-3: 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-4: Introduction, definition, derivation trees, simplification, Context-Free Grammar (CFG), Parse Trees.

Unit-5: 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

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester - V
Course Code: CC-11
Course Title: Software Engineering (6 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:
- 1. Basic knowledge and understanding of the analysis and design of complex systems.
- 2. Ability to apply software engineering principles and techniques.

Theory: 60 Lectures

Unit-1: Software Engineering Fundamentals: Definition of software product, Software Engineering Paradigms; Software engineering, Knowledge engineering, and End user development approaches.

Unit-2: 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-3: System Documentation: Principles of system documentation, types of documentation and their importance.

Unit-4: 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-5: 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-6: Role of CASE Tools: Relevance of CASE Tools, High-end and Low-end CASE Tools.

Unit-7: Coding and Programming: Choice of programming languages, mixed language programming and cell semantics, reengineering legacy systems, Coding standard.

Unit-8: 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-9: 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-10: Software Project Management: Planning software, projects, Work breakdown structures, Integrating software design and project planning, Software project teams, Projecting monitoring and control.

Reference Books:

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 - V
Course Code: CC-12
Course Title: Programming in Java (6 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.

Theory: 60 Lectures

Unit-1: Getting Started: Features of Java; Java Magic; Byte Code.

Unit-2: 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-3: 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-4: Arrays and Strings: The String Constructors, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, String Buffer.

Unit-5: 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-6: Exception Handling: Definition of an Exception; Exception Classes; Common Exceptions; Exception Handling Techniques.

Unit-7: Streams in Java: Abstract Streams; Stream Classes; Readers and Writers; Random Access Files; Serialization.

Unit-8: Event Handling: Components of an Event; Event Classes; Event Listener; Event-Handling; Adapter Classes; Inner Classes; Anonymous Classes.

Unit-9: JDBC: Database Management; Mechanism for connecting to a back end database; Loading the ODBC driver.

Unit-10: 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-11: 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-12 Swing: Concepts of Swing; Java Foundation Class (JFC); Swing Packages and Classes; Working with Swing- An Example; Swing Components.

Unit-13 Networking: Networking in Java; URL Objects.

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 change temperature from Celsius to Fahrenheit.
2. Write a Java program to change temperature from Fahrenheit to Celsius.
3. Write a Java program to find area and perimeter of a rectangle.
4. Write a Java program to find area and perimeter of a circle.
5. Write a Java Program to display whether a number is odd or even.
6. Write a Java Program to check if a number is Positive or Negative.
7. Write a Java program to find maximum of three numbers.
8. Write a Java program to swap two numbers.
9. Write a Java program to convert miles to kilometers.
10. Write a Java program to check whether a year is leap year or not.
11. Write a Java program to implement the concept of inheritance.
12. Write a Java program to show method overloading.
13. Write a Java program to show method overriding.
14. Write a Java program to show method hiding.
15. 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.
16. 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.

17. Write a Java program to explain "multilevel inheritance."
18. 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.
19. 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.
20. 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.
21. Write a program in Java to illustrate the use of interface in Java.
22. Write a Java program to show the use of all keywords for exception handling.
23. Write a Java program using try and catch to generate NegativeArrayIndex Exception and Arithmetic Exception.
24. Define an exception called "NoMatchFoundException" that is thrown when a string is not equal to "University". Write a program that uses this exception.
25. Write a Java program for calculating Factorial. Number should be taken through user input (Using Scanner, BufferedReader both).
26. Design a palindrome class that will input a string from console and check whether the string is palindrome or not.
27. Write a Java program to merge two strings.
28. Write a Java program in which total 4 threads should run. Set different priorities to the thread.
29. 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.
30. 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.
31. Design an applet that will display a text as scrolling marquee. The text can be changed by setting different "PARAMS" value.
32. Write a Java Program to Create a Banner Using Applet.
33. Write a Java Program to Display Clock Using Applet.

Semester - VI
Course Code: CC-13
Course Title: Internet Systems (6 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-1: 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-2: 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-3: File Transfer Protocol: Introduction to FTP, public domain Software, Types of FTP Servers, FTP clients, Common Commands.

Unit-4: Telnet: Telnet protocol, Server daemon, Telnet clients, Terminal emulation.

Unit-5: Usenet and Internet Relay Chat Introduction to World Wide Web: Evolution of WWW, Basics Features, WWW Browsers, WWW servers, HTTP & URL's.

Unit-6: 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-7: 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-8: 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-9: Interactivity Tools: CGI, ActiveX, VB Script and Java Script.

Unit-10: Multimedia and Graphics: VRML.

Unit-11: Search Engines: Technology overview, Popular Search Engines, How to register a Web site on search engines.

Unit-12: Internet Security: Overview of Internet Security threats, Firewalls, Introduction to AAA.

Unit-13: 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
3. The Practical Handbook of Internet Computing (Chapman & Hall/CRC Computer and Information Science Series) by Munindar P. Singh

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.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester - VI
Course Code: CC-14
Course Title: Intelligent Systems (6 Credit)
Course Objectives:
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:
- Solve basic AI based problems.
- Define the concept of Artificial Intelligence.
- Apply AI techniques to real-world problems to develop intelligent systems.
- Design AI application through LISP and Prolog.

Theory: 60 Lectures

Unit-1: Introduction: What is AI-importance of AI-objectives? Applications of AI in Natural Language Processing, Speech understanding, computer vision, planning etc.

Unit-2: Introduction to LISP: study of features and its application.

Unit-3: Knowledge and AI problem solving concepts: Its representation, organization – manipulation and acquisition, predicate calculus in AI – first order predicate logic & its use in knowledge representation – Resolution principle, use of resolution in reasoning and question answering, production systems and search strategies – production system and its variants – heuristic search methods.

Unit-4: Uncertainty Management: Fuzzy logic, Bayesian inferencing, certainty factor, Structured representation of knowledge – semantic networks, frames, conceptual dependency & scripts.

Unit-5: Learning: Learning automation, learning by induction, neural networks, and genetic algorithms.

Unit-6: Expert systems: Rule based system architecture-non production system architecture-knowledge acquisition methods-explanation methods-expert system shells.

Reference Books:
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.
5. Introduction to Prolog- R. P. Suri, Alpha Science
6. Prolog Programming for Artificial Intelligence-Ivan Bratko, Pearson
7. Lisp Programming-Rajeev Sangal, McGraw Hill Education India Pvt Ltd
8. Starting LISP for Artificial Intelligence (Artificial Intelligence Texts S.) – Coxhead. P., Alfred Waller Ltd.

Practical:

1. Write simple fact for the statements using PROLOG
2. Write a program in prolog to implement simple facts and Queries.
3. Write a prolog program to calculate the sum of two numbers.
4. Write a prolog program to calculate the factorial of a given number.
5. Write a prolog program to find the maximum of two numbers.
6. Write a prolog program to calculate the nth Fibonacci number.
7. Write a Prolog program to remove the Nth item from a list.
8. Write a program to solve 4-Queen problem.
9. Write a program in prolog to solve Tower of Hanoi
10. Write a program in prolog to solve 8 Puzzle problems
11. Write a program in prolog for Water jug problem.
12. Define a LISP function to compute sum of squares.
13. Define a LISP function to compute factorial of a given number.
14. Write a program to solve water jug problem using LISP
15. Solve any problem using depth first search.
16. Solve any problem using best first search.
17. Solve travelling salesman problem.
18. Write a Program to reverse the list.
19. Write Program to replace an integer from the list.

Semester – V
Course Code: DSE-01
Course Title: Introduction to Microprocessor and System Software (6 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-1: Introduction of Microcomputer System: CPU, I/O devices, clock, memory, bussed architecture, tristate logic, address bus, data bus and control bus.

Unit-2: Semiconductor Memories: Development of semiconductor memory, internal structure and decoding, memory read and write timing diagrams, MROM, ROM, EPROM, EEPROM, DRAM.

Unit-3: Architecture of 8-bit Microprocessor: Intel 8085 microprocessor, Pin description and internal architecture.

Unit-4: 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-5: 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-6: Assembly Language Programming: Assembler directives, simple examples; Subroutines, parameter, passing to subroutine.

Unit-7: 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-8: Interrupts: Interrupt structure of 8085 microprocessors, processing of vectored and non-vectored interrupts, latency time and response time; Handling multiple interrupts

Unit-9: 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-10: 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-11: 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-12: 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 4th bit of 8-numbers.
13. Write an assembly language program to perform addition of two 16-bit numbers using 8085 instruction set.
14. Write an assembly language program for subtracting of two 16-bit numbers by using-8085.
15. Write an assembly language program to perform multiplication of two 16-bit numbers using 8085 instruction set.

Semester - V
Course Code: DSE-02
Course Title: Data Warehousing and Data Mining (6 Credit)
Course Objectives:
This course gives an introduction to methods and theory for development of data warehouses and data analysis using data mining. Data quality and methods and techniques for pre-processing of data. Modelling and design of data warehouses. Algorithms for classification, clustering and association rule analysis.
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.
- 2. Understand KDD process for finding interesting pattern from warehouse.
- Remove redundancy and incomplete data from the dataset using data pre-processing methods.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Discover interesting patterns from large amounts of data to analyse for predictions and classification.
- Develop a data mining application for data analysis using various tools.

Theory: 60 Lectures

UNIT-1: 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-2: 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-3: 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-4: Association Rule Mining: Introduction - Mining single dimensional Boolean association rules from transactional databases - Mining multi-dimensional association rules.

Unit-5: Classification and Prediction: Classification Techniques - Issues regarding classification and prediction - decision tree - Bayesian classification –Classifier accuracy – Clustering – Clustering Methods - Outlier analysis.

Unit-6: Applications and Other Data Mining Methods: Distributed and parallel Data Mining Algorithms, Text mining- Web mining.

Reference Books:
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 - VI
Course Code: DSE-03
Course Title: .NET Technology (6 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–1: 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-1: 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-2: 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-3: 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-4: 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-5: Web Services: Introduction to webservice-Architecture of Web service: Universal Discovery Description and Integration-Web Service Description Language –Accessing webservice using different Clients.

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 : DSE-04
Course Title : Major Project & Viva-Voce (6 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.) (DSE-04) Project Proposal (Project's Title and Guide's Details)
IV	Guidelines for Project Formulation
V	Project Proposal Submission and Approval
VI	Project Report Formulation
VII	Important points while preparing the Project Report
VIII	List of Broad Areas of Application and Related Tools
IX	Certificate of Originality
X	Project Trainee Letter

I. MESSAGE FROM THE PROJECT CO-ORDINATOR

The BCA (Hons.) 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 (Hons.) programme.

The BCA (Hons.) 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 (Hons.) 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 (Hons.) 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. DSE-04 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 DSE-04 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. October to 31st October**
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 15th May**
4	Viva-Voce to be conducted	**Date Fixed by the University**

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

BANKURA UNIVERSITY
BANKURA, WEST BENGAL – 722 155

III. PROFORMA OF BCA (HONS.) PROJECT PROPOSAL (DSE-04)
(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. ……………………….……..…

4. **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)

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

6. **Software Used for this Project:**

………
…..……
…..……
…..……

Note:
 1. Use of Visual Basic and MS-Access as Front End and Back End respectively is forbidden. But, you are permitted to use Visual Basic with other Software. Also, you can use MS-Access with other software.
 2. Use of C or C++ Programming Language for Project Related to Database Management is strictly forbidden.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Signature of the Student   **Signature of the Guide**

Date: …………………..…..……   **Date: …………..…..…………**

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

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

For Office Use Only

………………………………………………………….

Signature, Designation, Stamp of the
Project Proposal Evaluator

Approved   **Not approved**   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

CBCS Syllabus for BCA (Hons.)

BANKURA UNIVERSITY
BANKURA, WEST BENGAL – 722 155

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 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.

V. PROJECT PROPOSAL SUBMISSION AND APPROVAL

After finalising 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
  - o Bibliography
  - o 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 (Hons.) 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

Note:
Projects should not be developed using the packages like Dbase, Foxpro, Visual Foxpro . Also, projects should not be developed using the combination of Visual Basic as the front end and MS-Access as the back end.

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 (Hons.), 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 …………………………..

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 (Hons.) Course Bankura

University and has to do a project in his/her final semester starting from January / July session. The project

is compulsory for BCA (Hons.) 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

---

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Semester - I
Course Code: GE-01
Course Title: Mathematics-I (6 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-1: Algebra: Sets, Union and Intersection, Complement, Mapping, Composition, notion of a Group, Ring, Field with simple examples.

Unit-2: Complex Number: Modulus and amplitude, De Moiver's theorem

Unit-3: 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 and Multiplication 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-4: Vector spaces, Subspaces, Bases and Dimensions, Co-ordinates, Linear Transformation, The Algebra of Linear Transformations.

Unit-5: 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-6: 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.

Reference Books:
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 - II
Course Code: GE-02
Course Title: Principals of Accounting and Costing (6 Credit)
Course Objectives:
This course exposes students to a broad range of accounting and cost accounting concepts and their terminology. Topics include Primary accounting books, financial statements, how to identify, measure, and accumulate direct and indirect costs.

Learning Outcomes:
After the completion of this course, the students will be able to:
- Prepare journal, ledger and trial balance.
- Prepare Final Accounts.
- Find different ratios.
- Understanding different costs.
- Perform different accounting problems in Tally.

Theory: 60 Lectures

Unit-1: Introduction of Accounting: Introduction, Basic Accounting concepts and Conventions, Double Entry Accounting, The Accounting Trail, Financial Statement and their Nature, The Accounting Equation.

Unit -2: Primary Books- Introduction, Golden Rules, cash Book, Secondary Books- posting techniques in the ledger, Trial Balance and Final Account, Trading A/c, Profit and Loss A/c and Balance Sheet- Preparation of Trial Balance and Final Account

Unit -3 Working Capital Management: Meaning, Classification, Importance, Working Capital Cycle, Factors determining Working Capital, Different Sources of Working Capital, Concept of working capital management.

Unit -4: Financial Statement Analysis: Meaning of Ratio Analysis, Balance sheet Ratio, Profit and Loss Account Ratio, Combined Ratio, Advantages and Limitations of ratio Analysis and Primary problems regarding ratios, Meaning of Fund Flow Statement, Objectives of Fund Flow Statement, Meaning of Cash Flow Statement, Objectives of Cash Flow Statement, Uses of Cash Flow Statement, Difference between Cash Flow and Fund Flow statement.

Unit-5: Understanding Cost: Introduction, Classification of Cost (on the basis of behaviour and on the basis of element), Overhead and non-cost items, Classification of Overhead, Determination of Total Cost, Cost Sheet, Preparation of Cost Sheet, Necessity of preparation of cost Sheet

Unit-6: Marginal Costing and Break-even Analysis: Introduction, Meaning and Features of Marginal Costing, Limitations, Profit Volume Ratio, Contribution, Break- even Point, Margin of Safety, Simple Problems on Marginal Costing.

Reference Books:
1. Accounting for managers – Asish K. Bhattacharya; PHI.
2. Financial accounting for management – N Ramachandran & am Kr. Kakni; Tata McGraw-Hill.
3. Modern Accountancy-Amitabha Mukherjee & Mohammed Hanif; Tata McGraw-Hill.
4. Financial Accounting – Ashok Banerjee; Excel Books.
5. Introduction to Financial Accounting – Horngren; Pearson Books.

6. Accounting & Finance for Managers – T.P. Ghosh; Taxmann.
7. Financial Management – I.M. Pandey; Vikas Publishing House.
8. Financial Management – Khan & Jain; Tata McGraw Hill.
9. **Fundamentals of Financial Management – Chandra Bose; PHI.**

Semester - III
Course Code: GE-03
Course Title: Mathematics-II (6 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-1: 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-2: 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-3: Integral Calculus: Rules of Integration of Indefinite Integrals, Solution of Definite Integrals and their elementary properties. Idea of improper integrals.

Unit-4: 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-5: Sequence and Series: Bounded and unbounded sequences, Convergence or divergence of a sequence, Behavior 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)

Reference Books:
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 - IV
Course Code: GE-04
Course Title: Mathematics-III (6 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-1: 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-2: Collection and presentation of data, Frequency distribution, Measures of central tendency, Measures of dispersion, Binomial, Poisson and Normal distribution.

Unit-3: Bivariate Frequency Distributions (scatter Diagram, Correlation coefficient and its properties, regression lines, correlation index and correlation ratio, rank correlation).

Unit-4: Multiple linear regression, multiple correlation, partial correlation (for 3 variables only).

Unit-5: Random sampling, expectations and standard error of sampling mean. Expectation and standard error of sampling proportions.

Unit-6: Test of significance based on t, F, and CHI square distribution.

Unit-7: 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-8: Integration: Trapizoidal and Simpson's 1/3 rules.

Unit-9: Solution of linear equations: Gaussian elimination, Gauss Sheidal method.

Unit-10: Solution of different equations; Eular's, Taylor's series, Runge-kutta (order-2)

Reference Books:
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: SEC-01A
Course Title: Android Programming (2 Credit)
Course Objectives:
The course is designed to provide complete knowledge of Android Programming. Students will be able to develop logics which will help them to create programs, applications through Android 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 Android applications.

Theory: 60 Lectures

Unit-1: Introduction: History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.

Unit-2: Overview of object oriented programming using Java: OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

Unit-3: Development Tools: Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project – Hello Word, run on emulator, Deploy it on USB-connected Android device.

Unit-4: User Interface Architecture: Application context, intents, Activity life cycle, multiple screen sizes.

Unit-5: User Interface Design: Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners (Combo boxes), Images, Menu, and Dialog.

Unit-6: Database: Understanding of SQLite database, connecting with the database.

Reference Books:
1. Android application development for java programmers. By James C. Sheusi.
2. Publisher: Cengage Learning, 2013.
3. Android Developer Tools Essentials by Mike Wolfson - O'Reilly Media Publications
4. Learn Java for Android Development, 2nd Edition - Jeff Friesen - Apress Publications
5. OpenGL ES 2 for Android - Kevin Brothaler- The Pragmatic Programmers
6. Android Application Development (With Kitkat Support), Black Book by Pradeep Kothari, Kogent Learning Solutions Inc.

Practical:

1. Install and Configure java development kit (JDK), android studio and android SDK.
2. Configure android development tools (ADT) plug-in and create android virtual device.
3. Develop a program to display Hello World on screen.
4. Develop a program to implement linear layout and absolute layout.
5. Develop a program to implement frame layout, table layout and relative layout.
6. Develop a program to implement Text View and Edit Text.
7. Develop a program to implement AutoComplete Text View.
8. Develop a program to implement Button, Image Button and Toggle Button.
9. Develop a program to implement login window using above UI controls.
10. Develop a program to implement Checkbox.
11. Develop a program to implement Radio Button and Radio Group.
12. Develop a program to implement Progress Bar.
13. Develop a program to implement List View, Grid View, Image View and Scroll View.
14. Develop a program to implement Custom Toast Alert.
15. Develop a program to implement Date and Time Picker.
16. Develop a program to create an activity.
17. Develop a program to implement new activity using explicit intent and implicit intent.
18. Develop a program to implement content provider.
19. Develop a program to implement service.
20. Develop a program to implement broadcast receiver.
21. Develop a program to access and implement sensors.
22. Develop a program to access the built in Camera.
23. Develop a program for providing Bluetooth connectivity.
24. Develop a program for animation.
25. Perform Sync task using SQLite.
26. Create sample application with login module. (Check username and password) On successful login, Change Text View "Login Successful" and on login fail, alert user using Toast "Login fail".
27. Create login application where you will have to validate username and password till the username and password is not validate, login button should remain disabled.
28. Develop a program to
    a. Send SMS
    b. Receive SMS
29. Develop a program to send and receive e-mail.
30. Deploy a map-based application.

Semester - III
Course Code: SEC-01B
Course Title: PHP Programming (2 Credit)
Course Objectives:
The course is designed to provide complete knowledge of PHP Programming. Students will be able to develop logics which will help them to create programs, applications through PHP 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 website by PHP.

Theory: 60 Lectures

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Unit-1: HTML Basics: HTML Introduction, History of HTML, HTML Elements, Attributes, HTML Headings, Paragraphs, HTML Formatting, Fonts, Styles, HTML Links, Images, Tables, HTML Lists, Forms, Frames, HTML Colors, Colornames, Colorvalues, HTML Quick List.

Unit-2: Introduction to PHP: Evaluation of PHP, Basic Syntax, Defining variable and constant, PHP Data type, Operator and Expression.

Unit-3: Decisions and loop: Making Decisions, Doing Repetitive task with looping, Mixing Decisions and looping with HTML.

Unit-4: Function: What is a function, Define a function, Call by value and Call by reference, Recursive function, String Creating and accessing, String Searching & Replacing String, Formatting String, String Related Library function.

Unit-5: Array: Anatomy of an Array, Creating index based and Associative array Accessing array, Element Looping with Index based array, Looping with associative array using each () and foreach(), Some useful Library function.

Unit-6: Handling HTML Form with PHP: Capturing Form, Data Dealing with Multi-value filed, and Generating File uploaded form, redirecting a form after submission.

Unit-7: Working with file and Directories: Understanding file& directory, Opening and closing, a file, Coping, renaming and deleting a file, working with directories, Creating and deleting folder, File Uploading & Downloading.

Unit-8: Session and Cookie: Introduction to Session Control, Session Functionality What is a Cookie, Setting Cookies with PHP. Using Cookies with Sessions, Deleting Cookies, Registering Session variables, Destroying the variables and Session.

Unit-9: Database Connectivity with MySQL: Introduction to RDBMS, Connection with MySQL Database, Performing basic database operation (DML) (Insert, Delete, Update, Select), Setting query parameter, Executing query-Join (Cross joins, Inner joins, Outer Joins, Self joins.), handling errors, debugging and diagnostic functions, validating user input through Database layer and Application layer, formatting query output with Character, Numeric, Date and time, sample database applications.

Unit-10: Exception Handling: Understanding Exception and error, Try, catch, throw. Error tracking and debugging.

Reference Books:
1. Ivan Bayross, Web Enabled Commercial Application Development Using Html, Dhtml, javascript, Perl Cgi , BPB Publications, 2009.
2. Learning PHP, MySQL, books by ' O' riley Press
3. VIKRAM VASWANI, —PHP and MySQL , Tata McGraw-Hill
4. BEN FORTA , MySQL Crash course — SAMS

Practical:

1) Write a PHP program to print Sum of Digits
2) Write a PHP program to print Even or odd number
3) Write a PHP program to print Prime numbers between 100 - 250
4) Write a PHP program to print Table of number
5) Write a PHP program to print factorial of a number.
6) Write a PHP program to check Armstrong number.
7) Write a PHP program to check palindrome number.
8) Write a PHP program to print Fibonacci series without using recursion and using recursion.
9) Write a PHP program to reverse given number.
10) Write a PHP program to reverse given string.
11) Write a PHP program to swap two numbers with and without using third variable.
12) Design Student Sign-Up Form Using HTML, JavaScript, HTML5 & CSS

    Procedure:

        a) Create HTML Page named as "sign_up.html" and add the following tags details

- Form Tag, Paragraph Tag
- Different Heading Tag, Line Break, Horizontal line
- Input tag for text box and submit button
- Use Script tag for validation of input data
- Use HTM5 attribute for validate the input data
- Use Style Tag for creating internal CSS, Inline CSS
- Use Link Tag to refer an external CSS

        b) Save given page with sign_up.html by choosing 'All files' from Save as Type in any respective folder

        c) Execute the page by double clicking on name of page from respective folder, It will show result on particular browser (eg Mozilla, Chrome, Internet Explorer)

13) Design the personal information form, submit and retrieve the form data using php $_POST, $_GET, $ REQUEST variable

    Procedure:

        a) Create HTML page named as 'test.html' in DW site Folder for taking input from user and add following tags in it

- Use form tag with method 'post' & action 'test1.php' within html code
- Take Input tags for creating text box, radio button,check box and submit button
- Use textarea tag for inputting address
- Use select & option tag for creating list

        b) Create PHP page named as test1.php in given folder for retrieving and displaying test.html data

        c) Store submitted data in php variable as $fn=$_POST["fname"]; $g=$_POST["gender"] etc

        d) Display These Variable Using echo function 5) Save & Execute Php File In web Browser

14) Design a webpage to display your college with hyperlink.
15) Create Log-in page with validation.

Semester - III
Course Code: SEC-01C
Course Title: Digital Image Processing (2 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.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

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-1: 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-2: 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-3: 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.

Practical

1. Write a program (in MATLAB) to read an image and display its property.
2. Write a program (in MATLAB) to display an image.
3. Write a program (in MATLAB) to write an image variable as image.
4. Write a program (in MATLAB) to enlarge an image to its double size.
5. Write a program (in MATLAB) to rotate an image in clockwise and anticlockwise direction.
6. Write a program (in MATLAB) to convert and rgb image to gray scale image.
7. Write a program (in MATLAB) to implement the Basic Gray Level Transformation.
- Image Negative
- Log Transformation
- Power Law Transformation
- Piecewise Linear Transformation (Contrast Stretching)
8. Write a program (in MATLAB) to generate Histogram for an Image and plot histogram in various ways (imhist, bar, stem, plot).
9. Write a program (in MATLAB) to perform Histogram Equalization.
10. Write a program (in MATLAB) to implement Arithmetic and Logical operation:
- Image Subtraction
- Image Averaging
11. Write a program (in MATLAB) to implement Smoothing Spatial Filters:
- Linear filter (Standard Average/BOX, Weighted Average)

- Ordered Statistic (Median)

12. Write a program (in MATLAB) to implement Sharpening Spatial Filters:
    - Laplacian
13. Write a program (in MATLAB) to implement Smoothing (Lowpass) Frequency Domain Filters.
14. Write a program (in MATLAB) to implement Sharpening (Highpass) Frequency Domain Filters.
15. Write a program (in MATLAB) to implement Homomorphic filter.
16. Write a program (in MATLAB) to implement different Noise model:
    - Uniform noise
    - Gaussian noise
    - Salt & Pepper noise
    - Rayleigh noise
    - Exponential noise
    - Rrlang noise
17. Write a program (in MATLAB) to implement Restoration in the presence of noise only - Spatial Filtering:
    - Arithmetic mean filter
    - Geometric mean filter
    - Harmonic mean filter
    - Median filter
    - Max filter
    - Min filter
18. Write a program (in MATLAB) to implement Color conversion for RGB model to HIS model.
19. Write a program (in MATLAB) to implement Color conversion for HIS model to RGB model.
20. Write a program (in MATLAB) to perform smoothing and sharpening on Color image.

Semester - IV
Course Code: SEC-02A
Course Title: Python Programming (2 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-1: 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-2: 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-3: 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-4: 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-5: 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-6: 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-7: 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-8: 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-9: Python OOP: Object oriented programming and classes in Python - creating classes, instance objects, accessing members, Data hiding (the double underscore prefix), Built-in class attributes, Garbage collection: the constructor, Overloading methods and operators, Inheritance - implementing a subclass, overriding methods, Recursive calls to methods, Class variables, class methods, and static methods,

Unit-10: 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-11: 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 - IV
Course Code: SEC-02B
Course Title: Wireless Mobile Communication (2 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-1: Introduction to mobile communication and computing: The application and significance of mobile communications, mobile and wireless devices along with the history of wireless communication, simple Reference Model of communication.

Unit-2: Wireless Transmission – I: various frequencies used for communication, types of signals and the antennas used for communication, methods of signal propagation and the techniques of multiplexing, learn how signals are propagated using various modulation techniques - analog modulation and digital modulation, spread Spectrum technology like Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS), cellular systems used for mobile communications along with the way frequency and planned.

Unit-3: Wireless Medium Access Control: reasoning of need for a Specialized MAC in wireless domain, various medium accessing technique viz. Space Division Multiple Access (SDMA), Frequency Division Multiple Access (FDMA), Time Division Multiple Access (TDMA) and Code Division Multiple Access (CDMA).

Unit-4: Wireless Telecommunication Systems: Global systems used for Mobile (GSM) Communications, architecture of a GSM system and GSM –TDMA/ FDMA frame and various types of logical channels in GSM system and the use of GSM hierarchy of frames GSM protocol layers for signaling, High Speed Circuit Switched Data (HSCSD) and General Packet Radio Service (GPRS).

Unit-5: Universal Mobile Telecommunication System (UMTS): Universal Mobile Telecommunication System (UMTS) where UMTS system architecture and UMTS radio interface, UTRA Network (UTRAN).

Unit-6: Wireless LANs: Various mobile communication technologies according to IEEE, characteristics of wireless LANs and the comparison of infrared and radio transmission technologies, infrastructure-based wireless networks along with the Ad hoc wireless networks. Architecture of an infrastructure based IEEE 802.11 and architecture of IEEE 802.11 ad hoc network, various IEEE standards of 802.11.

Unit-7: Mobile Network Layer: Mobile IP technology, goals, assumptions and requirements of Mobile IP, techniques and various entities and terminologies of mobile IP. Dynamic Host Configuration Protocol (DHCP).

Unit-8: Mobile Transport Layer: Traditional TCP protocol and the need of modifying this protocol in wireless domain, how the classical TCP is improved into Indirect TCP, Snooping TCP and Mobile TCP, other methods of improving the TCP for wireless domain, Fast retransmit/fast recovery, Transmission/ time-out freezing, Selective retransmission and Transaction oriented TCP as the improvements for TCP in wireless domain, TCP Over 2.5/3G wireless networks.

Unit-9: Data Processing and Mobility: Effect of mobility of the mobile nodes and the way data are categorized and managed when the nodes are mobile, way transactions are managed in Mobile Database Systems. Various execution models for mobile database system.

Unit-10: Mobile Ad hoc Networks: Mobile Ad hoc Networks (MANETs), properties of a MANET and the spectrum of MANET applications. Various routing algorithms used in MANETs, security aspects in MANETs.

Unit-11: 4G Technology: 4G technology and its various flavors, WiMAX (Worldwide Interoperability for Microwave Access), LTE (Long Term Evolution) and HSPA+ (High Speed Packet Access) technology.

Unit-12: HTML5 on Mobile Devices: HTML5 technology required for mobile websites and mobile applications on mobile operating systems, tools of HTML5 which are very relevant for developing the web applications for the mobile devices.

Reference Books:
1. Upena Dalal, Manoj K. Shukla, Wireless and Mobile Communication, Oxford Higher Education
2. Upena Dalal, Wireless Communication and Networks, Oxford Higher Education
3. JOCHEN H SCHILLER, MOBILE COMMUNICATIONS, 2ED, Pearson
4. Sanjay Sharma, Wireless Communication, S. K. Kataria & Sons
5. Rappaport , Wireless Communications: Principles and Practice, 2e, Pearson

Practical:
1. Design an Amplitude Modulator using Transistor.
2. Design and Study Envelope Detector for Demodulation of AM Signal.
3. Design and Study FM Generator and Detector Circuit.
4. Design & Study Pulse Amplitude Modulation (PAM).
5. Design & Study Pulse Width Modulation (PWM).
6. Design & Study Pulse Position Modulation (PPM).
7. Design & Study ASK, PSK and FSK Modulators.
8. Understanding Cellular Fundamentals like Frequency Reuse, Interference, cell splitting, multi path environment, Coverage and Capacity issues using communication software.
9. Knowing GSM and CDMA architecture, network concepts, call management, call setup, call release, Security and Power Control, Handoff Process and types, Rake Receiver etc.
10. Study of GSM handset for various signaling and fault insertion techniques (Major GSM handset sections: clock, SIM card, charging, LCD module, Keyboard, User interface).
11. To study transmitters and receiver section in mobile handset and measure frequency band signal and GMSK modulating signal.
12. To study various GSM AT Commands their use and developing new application using it. Understating of 3G Communication System with features like; transmission of voice and video calls, SMS, MMS, TCP/IP, HTTP, GPS and File system by AT Commands in 3G network.
13. Study of DSSS technique for CDMA, observe effect of variation of types of PN codes, chip rate, spreading factor, processing gains on performance.
14. To learn and develop concepts of Software Radio in real time environment by studying the building blocks like Base band and RF section, convolution encoder, Interleaver and DeInterleaver.

15. To study and analyze different modulation techniques in time and frequency domain using SDR kit.

Semester - IV
Course Code: SEC-02C
Course Title: Database Programming with PL/SQL (2 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-1: Introduction to PL/SQL: History, Benefits, Creating PL/SQL Blocks

Unit-2: Variables and Data types: Defining Variables and Data types, 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-3: SQL in PL/SQL: Review of SQL DML, Retrieving Data in PL/SQL, Manipulating Data in PL/SQL, Transaction Control Statements.

Unit-4: 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-6: Cursors and Parameters: Introduction to Explicit Cursors, Explicit Cursor Attributes, Cursor FOR Loops, Cursors with Parameters, Cursors for UPDATE, Multiple Cursors.

Unit-7: Exception Handling: Handling Exceptions, Trapping Oracle Server Exceptions, Trapping User-Defined Exceptions, Recognizing the Scope of Exceptions.

Unit-8: Procedures: Creating Procedures, Parameters in Procedures, Passing Parameters.

Unit-9: 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-10: Packages: Creating Packages, Managing Package Concepts, Advanced Package Concepts, Persistent State of Package Variables, Oracle-Supplied Packages.

Unit-11: Triggers: Introduction, Creating DML Triggers, Creating DDL and Database Event Triggers, Managing Triggers.

Unit-12: Dependencies: Introduction to Dependencies, Understanding Remote Dependencies.

BANKURA UNIVERSITY

CBCS Syllabus for BCA (Hons.)

Unit-13: 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.

========== XXX =========