BCA (Bachelor of Computer Application)

Course Objectives Outcome (COS)

Course: BCA103T - Problem Solving Techniques Using C

PROGRAMME: **Computer Application**	COURSE: **Problem Solving Techniques Using C**
DEGREE:**BCA**	SEMESTER: **1** CREDITS: **2**
COURSECODE: **BCA 103 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **4 (weekly)**

Course pre-requisites:

Basic understanding of techniques used in Programming, algorithm and Flowchart.

Course Objectives

1. Learn the difference between software and hardware.
2. Learn data types, keywords and control structures of C
3. Learn to write algorithms for the given Problems.
4. Learn to write Flowcharts for the given Problems.
5. Learn to map problems to programming features of C.
6. Learn the looping concept.
7. Learn the concept of functions.
8. Learn to write good portable C programs.

Course Outcomes

Upon successful completion of the course, a student will be able to:

Co.1:- Appreciate and understand the differences between hardware and software.
Co.2:- Analyze a given problem and develop an algorithm to solve the problem.
Co.3:- Improve upon a solution to a problem.
Co.4:-Use the 'C' language constructs in the right way.
Co.5:-Design, develop and test programs written in 'C'.

Assessment Methodologies

SL .NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA103P C-Programming Lab

PROGRAMME: **Computer Application**	COURSE: **C-Programming Lab**
DEGREE:**BCA**	SEMESTER: **1** CREDITS: **2**
COURSECODE: **BCA 103 P**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE:PRACTIVCAL	CONTACT HOURS: **3 hr(weekly)**

Course: BCA104 Digital Electronics

PROGRAMME: **Computer Application**	COURSE: **Digital Electronics**
DEGREE:**BCA**	SEMESTER: **1** CREDITS: **2**
COURSECODE: **BCA 104 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **4 (weekly)**

Course pre-requisites;

Basic understanding of Number systems and Basic Electronics concepts.

Course Objectives

The objective of this course is to introduce the organization of a computer and its principal components, viz, ALU, Control, Memory and Input/output. The course will also enable the student to understand the design components of a digital subsystem that required realizing various components such as ALU, Control, etc.

Course Outcomes:

Upon successful completion of the course, a student will be able to:

Co.1:-An ability to understand theory of Digital Design and Computer Organization to provide an insight of how basic computer components are specified.
Co.2:-An ability to understand the functions of various hardware components and their building blocks
Co.3:- An ability to understand and appreciate Boolean algebraic expressions to digital design
Co.4:-An in depth understanding of sequential! Combinational circuits
Co.5:-An in depth understanding of realization of different combinational/sequential circuits
Co.6:- An in depth understanding of different stages of an instruction execution
Co.7:-An in depth understanding of how different hardware components are related and works in coordination
Co.8:-An ability to understand computer buses and input/output peripherals

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA104P

PROGRAMME: **Computer Application**	COURSE: **Digital Electronics**
DEGREE:**BCA**	SEMESTER: **1** CREDITS: **1**
COURSECODE: **BCA 104 P**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Practical's**	CONTACT HOURS: **3 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA104P**

Course: BCA105T -Discrete Mathematics

PROGRAMME: **Computer Application**	COURSE: **Discrete Mathematics**
DEGREE:**BCA**	SEMESTER: **1** CREDITS: **3**
COURSECODE: **BCA 105 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Practicals**	CONTACT HOURS: **3 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA104P**

Course Objectives

To develop logical thinking and its application to computer science (to emphasize the importance of proving statements correctly and de-emphasize the hand-waving approach towards correctness of an argument). The subject enhances one's ability to reason and ability to present a coherent and mathematically accurate argument. About 40% of the course time will be spent on logic and proofs and remaining 60% of the course time will be devoted to functions, relations, etc.

Prerequisites

PHI 251 and MAT 295

Course Outcomes

After completing this course satisfactorily, a student will:

Co.1:-Be able to construct simple mathematical proofs and possess the ability to verify them ABET[(a, j)].

Co.2:-Have substantial experience to comprehend formal logical arguments ABET[(a, b, c)].

Co.3:-Be skillful in expressing mathematical properties formally via the formal language of propositional logic and predicate logic ABET[(a)].

Co.4:-Be able to specify and manipulate basic mathematical objects such as sets, functions, and relations and will also be able to verify simple mathematical properties that these objects possess ABET[(a)].

Co.5:-Acquire ability to describe computer programs (e.g. recursive functions) in a formal mathematical manner ABET[(a, c, i, j)]

Co.6:-Be able to apply basic counting techniques to solve combinatorial problems ABET[(a)].

Co.7:-Gain experience in using various techniques of mathematical induction (weak, strong and structural induction) to prove simple mathematical properties of a variety of discrete structures ABET[(a, c, j)].

Outcome Measurement

The course outcomes will be mainly measured via in-class exams, homework's, quizzes. In addition, lab assignments to restress mathematical concept will be used.

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA203T Data Structures

PROGRAMME: **Computer Application**	COURSE: **Data Structures**
DEGREE:**BCA**	SEMESTER: **2** CREDITS: **2**
COURSECODE: **BCA 203 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **4 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA203P**

Course pre-requisites:

Sound programming knowledge in C and algorithms.

Course Objectives

1. Notion of Abstract Data Types (ADT) & Recursive access on them
2. Relation between Data Structure operations and Amortized Complexity analysis

3. How to implement Iterated Lists and variations thereof
4. Tree data structures and how to balance them, for specific access needs
5. Understanding Graph representations, Event modeling, spatial and temporal relational data
6. Choose a Data structure, a set of access methods and determine their asymptotic efficiency

Course Outcomes

Upon successful completion of the course student should be able to:

Co.1:-Analyze data structure impact on algorithms, program design and program performance.
Co.2:-Understand and apply amortized analysis on data structures, including binary search trees, heaps, and disjoint sets.
Co.3:- Explain & describe the applications of static and dynamic trees.
Co.4:- Design, implement, and use advanced ADTs.

Course Outcomes

Upon successful completion of the course, a student will be able to:
Co.1:- Appreciate and understand the differences between hardware and software.
Co.2:- Analyze a given problem and develop an algorithm to solve the problem
Co.3:- Improve upon a solution to a problem
Co.4:- Use the 'C' language constructs in the right way
Co.5:- Design, develop and test programs written in 'C'

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA203P

PROGRAMME: **Computer Application**	COURSE: **Data Structures Lab**
DEGREE:**BCA**	SEMESTER: **2** CREDITS: **1**
COURSECODE: **BCA 203 P**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Practical's**	CONTACT HOURS: **3 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA203P**
LABCOURSE NAME: Data Structures Lab	CONTACT HOURS: 3P/WEEK

Course: BCA 204 Data Base Management System

PROGRAMME: **Computer Application**	COURSE: **Data Structures Lab**
DEGREE:**BCA**	SEMESTER: **2** CREDITS: **1**
COURSECODE: **BCA 204T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **3 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA204p**

Course pre-requisites:

Fundamental concepts about data and basic programming knowledge.

Course Objectives

The objective of the course is to enable students to understand and use a relational database system. Introduction to Databases, Conceptual design using ERD, Functional dependencies and Normalization, Relational Algebra is covered in detail. Students learn how to design and create a good database and use various SQL operations. The course concludes with an overview of transaction management and introduction to advanced and non-relational databases.

Course Outcomes

Co.1:- Able to master the basic concepts and understand the applications of database systems.
Co.2:- Able to construct an Entity-Relationship (E-R) model from specifications and to transform to relational model.
Co.3:- Able to construct unary/binary/set/aggregate queries in Relational Algebra.
Co.4:- Understand and apply database normalization principles.
Co.5:- Able to construct SQL queries to perform CRUD operations on database. (Create, Retrieve, Update, Delete)
Co.6:- Understand principles of database transaction management, database recovery, security.

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA 204P Database Lab (Oracle)

PROGRAMME: **Computer Application**	COURSE: **Data Structures Lab**
DEGREE:**BCA**	SEMESTER: **2** CREDITS: **1**
Lab COURSECODE: **BCA 204P**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Practicals**	CONTACT HOURS: **3 (weekly)**

Course: BCA 205T Numerical Analysis & Statistical Methods

PROGRAMME: **Computer Application**	COURSE: **Numerical Analysis & Statistical Methods**
DEGREE:**BCA**	SEMESTER: **2** CREDITS: **3**
COURSECODE: **BCA 205T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **5 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA204p**

Course pre-requisites

Basic knowledge of formulation of algorithms.

Course Objectives

The primary goal is to provide engineering majors with a basic knowledge of numerical methods including: root- finding, elementary numerical linear algebra, integration, interpolation, solving systems of linear equations, curve fitting, and numerical solution to ordinary differential equations. ‘C’ language software environment used for implementation and application of these numerical methods. The numerical techniques learned in this course enable students to work with mathematical models of technology and systems.

Course Outcomes

Co.1:- An ability to apply knowledge of mathematics, science, and engineering.
Co.2:- An ability to design and conduct experiments, as well as to analyze and interpret data.
Co.3:- An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
Co.4:- An ability to function on multidisciplinary teams

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA303T Object Oriented Programming using C++

PROGRAMME: **Computer Application**	COURSE: Object Oriented Programming using C++
DEGREE:**BCA**	SEMESTER: **3** CREDITS: **2**
COURSECODE: **BCA 303 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **4 (weekly)**
CORRESPONDING LABCOURSE CODE	(IFANY): **BCA303P**

Course pre-requisites

Fundamental programming knowledge about C language.

Course Objectives

The fundamental point in learning programming is to develop the critical skills of formulating programmatic solutions for real problems. It will be based on basic knowledge of algorithms and procedural programming language. Once the basic skill of writing programs using loop, methods and arrays will be clear then the student can develop object oriented software using class encapsulation and inheritance.

To impart the basic concepts of Java Programming and to develop understanding about Basic Object oriented Design.

Course Outcomes

Co.1:- Understand fundamental constructs of OOP.
Co.2:- Get the knowledge of UML with skills to draw UML diagrams.
Co.3:- Get the knowledge of different forms of OO Implementation.
Co.4:- Apply object oriented programming concepts in problem solving through C++.
Co.5:- Design and implement Applet and event handling mechanisms in programs

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests Direct	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA303P - OOP using C++ Lab

PROGRAMME: **Computer Application**	COURSE: OOP in C++ Lab
DEGREE:**BCA**	SEMESTER: **3** CREDITS: **1**
COURSECODE: **BCA 303 P**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Practical's**	CONTACT HOURS: **3 (weekly)**

Course: BCA 303T -Operating System

PROGRAMME: **Computer Application**	COURSE: OOP in C++ Lab
DEGREE:**BCA**	SEMESTER: **3** CREDITS: **3**
COURSECODE: **BCA 303 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **5hr (weekly)**

Course pre-requisites:

Basic knowledge of working principle of computer. Familiarity with working with WINDOWS operating system.

Course Objectives

1. To understand the services provided by and the design of an operating system.
2. To understand the system programs, system components.
3. To understand the structure and organisation of the file system.
4. To understand what a process is and how processes are synchronized and scheduled.
5. To understand different approaches to memory management.
6. Students should be able to use system calls for managing processes, memory and the file system.
7. Students should understand the data structures and algorithms used to control deadlock.
8. To understand disk scheduling, protection and security.

Course Outcomes

Co.1:- Analyse the concepts of processes in operating system and illustration of the scheduling of processor for a given problem instance.

Co.2:- Understand the evolution of operating system.

Co.3:- Identify the dead lock situation and provide appropriate solution so that protection and security of the operating system is also maintained.

Co.4:- Analyze memory management techniques, concepts of virtual memory and disk scheduling.

Co.5:- Understand the implementation of file systems and directories along with the interfacing of IO devices with the operating system.

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA 304T -Financial Accounting & Management

PROGRAMME: **Computer Application**	COURSE: **Financial Accounting & Management**
DEGREE:**BCA**	SEMESTER: **3** CREDITS:**2**
COURSECODE: **BCA 304 T**	COURSE AREA/DOMAIN: **NA**
COURSE TYPE: **Theory**	CONTACT HOURS: **4hr (weekly)**

Course pre-requisites:

Fundamental concepts about account.

Course Objectives

The objective of the course is to strengthen the fundamentals of accounting and provide strong foundation for other accounting courses. The course will intensify knowledge on all the basic components by using double entry book keeping perspective.\

Course Outcomes

The students will be able to:

Co.1:- Define fundamental accounting concepts, Conventions & terminologies.

Co.2:- Describe the importance, functions & objectives of books of entry, subsidiary books, bank reconciliation statement and Final accounts.

Co.3:- Prepare books of entry, subsidiary books, bank reconciliation statement and Final accounts using double entry book keeping.

Co.4:- To rectify the errors located in books of entry & subsidiary books.

Assessment Methodologies

SL.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: **Financial Accounting & Management**	Semester : 3 CREDITS: 1
COURSECODE: BCA 304P	COURSE TYPE: Theory
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 4 (weekly)
CORRESPONDING LABCOURSE CODE (IFANY): BCA 304P	LABCOURSE NAME: TALLY Lab

Course: BCA 403 T Visual Programming

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: Visual Programming	Semester : 4 CREDITS: 2
COURSECODE: BCA 403T	COURSE TYPE: Theory
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 4 (weekly)
CORRESPONDING LABCOURSE CODE (IFANY): BCA 403P	LABCOURSE NAME: Visual Programming Lab

Course pre-requisites:

Basic knowledge of Programming and basic user interface design.

Course Objectives

1. Learn basics of visual Basic programming
2. Learn how to design a given problem
3. Learn to use various paradigms of programming and user interface designing.
4. Learn Visual Basic as a programming language
5. Learn how to implement data structures and functions available in Visual Basic to solve problems

6. To explore the Microsoft Foundation Class programming concepts

Course Outcomes

After this course, the student will be able to

CO1. Analyze a given problem and implement an algorithm to solve the problem

CO2. Improve upon a solution to a problem

CO3. Implement the Visual Basic language constructs in the right way

CO4. Design, develop and test Applications written in Visual Basic.

CO5: Implement and innovate commands using the basic tool kit.

CO6: Develop the practice of writing windows applications through Object Oriented concepts

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests	Direct
3	University Examinations	Direct
4	Student Feed Back	Indirect

Course: BCA 403P Visual Programming Lab

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: Visual Programming	Semester : 4 CREDITS: 1
COURSECODE: BCA 403 P	COURSE TYPE: Practical
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 3 (weekly)

Course: BCA 404 UNIX and Shell Programming

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: Unix and Shell Programming	SEMESTER: 4 CREDITS: 2
COURSECODE: BCA 404T	COURSE TYPE: Theory
COURSE AREA/DOMAIN: Operating system	CONTACT HOURS: 4 (weekly)
CORRESPONDING LABCOURSE CODE: BCA404P	LABCOURSE NAME: Unix Lab

Course pre-requisites

Basic knowledge about Operating System.

Course Objectives

1. The aim of this course is to make you aware of the functioning of a multi-user operating system.
2. This course serves as a foundation course for other higher level course in UNIX.
3. The course will help you to learn commands while doing practical and it emphasizes more on those switches/options and flags which are most frequently used in real life.
4. To understand Unix Operating System
5. To explore the Basic Shell Commands

Course Outcomes:

After this course, the student will be able to

CO1: Discuss the architecture, networking and basic commands of UNIX. (Understand)

CO2: Implement various file processing commands used in UNIX. (Apply)

CO3: Apply Regular expression to perform pattern matching using utilities like grep, sed and awk. (Apply)

CO4: Construct various shell scripts for simple applications. (Apply)

CO5: Explain the process management using system calls UNIX environment (Understand)

CO6: Implement and innovate commands using the basic tool kit

CO7: Develop shell programs in vi/vim editor.

Assessment Methodologies

SL. NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA404 P UNIX Lab

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: Unix Lab	SEMESTER: 4 CREDITS: 1
COURSECODE: BCA404P	COURSE TYPE: Practical
COURSE AREA/DOMAIN: NA	CONTACT HOURS: 6 P/WEEK

Course: BCA 501T - : Data Communication and Networks

PROGRAMME: Computer Application	**COURSE TYPE:** Theory
DEGREE : BCA	**COURSE AREA/DOMAIN**: NA
COURSE : Data Communication And Networks	**CONTACT HOURS**: 4 (weekly)
SEMESTER: 5	**CORRESPONDING LABCOURSE CODE (IFANY):** NA
CREDITS: 3	

Course pre-requisites:
Fundamental knowledge about analog and digital communication. Basic knowledge about computer network and communication.

Course Objectives
1. To provide an introduction to the fundamental concepts on data communication and the design of computer networks.
2. To get familiarized with the basic protocols of computer networks and Networks Standards like LAN, MAC etc.

Course Outcomes
After this course, the student will be able to

Co.1:- Identify the different components in a Communication System and their respective roles.
Co.2:- Describe the technical issues related to the local Area Networks
Co.3:- Identify the common technologies available in establishing LAN infrastructure.
Co.4:- Understand computer network basics, network architecture, and TCP/IP and OSI reference models.
Co.5:- Identify and understand various techniques and modes of transmission
Co.6:- Describe data link protocols, multi-channel access protocols and IEEE 802 standards for LAN
Co.7:- Describe routing and congestion in network layer with routing algorithms and classify IPV4 addressing scheme
Co.8:- Discuss the elements and protocols of transport layer
Co.9:- Understand network security and define various protocols such as FTP, HTTP, Telnet, and DNS.
Co.10:- Understand about Telephone network.
Co.11:- Understand about Peer to peer communication, ARQ protocols, LAN, MAC and LAN Standards. Packet Switching Network

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA 502: Software Engineering

PROGRAMME: Computer Application	**COURSE TYPE:** Theory
DEGREE : BCA	**COURSE AREA/DOMAIN**: NA
COURSE : Software Engineering	**CONTACT HOURS**: 4 (weekly)
SEMESTER: 5	**CORRESPONDING LABCOURSE CODE (IFANY):** NA
CREDITS: 3	

Course pre-requisites:

Basic knowledge about Software Development life cycle.

Course Objectives

To develop an understanding of software engineering, software crisis, SDLC. Understanding the concept of software project planning – feasibility analysis, requirement analysis, SRS documents. Come to know the software designing strategies – structured analysis, structured design, DFD, structure chart. Understand concept of Project Management along with software testing, maintenance, back-up..

Course Outcomes

Co.1:- Evaluate and analyze the SDLC and basic architecture SRS documents.
Co.2:- Help to understand the software design and coding techniques.
Co.3:- Understand the software testing principles.
Co.4:- Understand the concept project management.
Co.5:- Identify various concepts of Advanced UML techniques.

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA503 T - : Computer Architecture

PROGRAMME: Computer Application	**COURSE TYPE:** Theory
DEGREE: BCA	**COURSE AREA/DOMAIN:** NA
COURSE: Computer Architecture	**CONTACT HOURS**: 4 (weekly)
SEMESTER: 5	**CORRESPONDING LABCOURSE CODE (IFANY):** NA
CREDITS: 3	
COURSECODE: BCA 503T	**LABCOURSE NAME:** NA

Course pre-requisites

Basic knowledge of Digital electronics.

Course Objectives

The objective of this course is to introduce the organization of a computer and its principal components, viz, ALU, Control, Memory and Input/output. The course will also enable the student to understand the design components of a digital subsystem that required realizing various components such as ALU, Control, CPU, IOP, Memory etc.

Course Outcomes

Upon successful completion of the course, a student will be able to:

Co.1:- An ability to understand theory of Digital Design and Computer Organization to provide an insight of how basic computer components are specified.

Co.2:- An ability to understand the functions of various hardware components and their building blocks.
Co.3:- An ability to understand and appreciate Boolean algebraic expressions to digital design.
Co.4:- An in depth understanding of sequential! Combinational circuits.
Co.5:- An in depth understanding of realization of different combinational/sequential circuits.
Co.6:- An ability to understand computer buses and input/output peripherals.
Co.7:- An ability to understand memory hierarchy and design of primary memory and DMA.
Co.8:- An ability to understand digital components in organisation like logic gates, combinational circuits, Flip Flop and sequential circuits.
Co.9:- An ability to understand digital components.
Co.10:- An ability to understand data representation and various binary codes.
Co.11:- An ability to understand designs an elementary basic computer.
Co.12:- An ability to understand organization and architecture of central processing unit.
Co.13:- An ability to understand the organization and architecture of IOP.

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA 504 T Java Programming

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: Visual Programming	Semester : 4 CREDITS: 2
COURSECODE: BCA 504 T	COURSE TYPE: Theory
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 4 (weekly)
CORRESPONDING LABCOURSE CODE (IFANY): BCA 504 P	LABCOURSE NAME: Java Programming Lab

Course pre-requisites:
Basic knowledge about C, C++.

Course Objectives

The fundamental point in learning programming is to develop the critical skills of formulating programmatic solutions for real problems. It will be based on basic knowledge of algorithms and procedural programming language. Once the basic skill of writing programs using loop, methods and arrays will be clear then the student can develop object oriented software using class encapsulation and inheritance.

To impart the basic concepts of Java Programming and to develop understanding about Basic Object oriented Design using UML and Applet.

Course Outcomes

Co.1:- Understands fundamental constructs of OOP.

Co.2:- Get the knowledge of UML with skills to draw UML diagrams.

Co.3:- Gets the knowledge of different forms of OO Implementation.

Co.4:- Apply object oriented programming concepts in problem solving through JAVA.

Co.5:- Design and implement Applet and event handling mechanisms in programs

Co.6:- Ability to create packages and interfaces.

Co.7:- Ability to implement error handling techniques using exception handling

Assessment Methodologies

S.NO	**DESCRIPTION**	**TYPE**
1	Student Assignment	Direct
2	Tests	Direct
3	University Examinations	Direct
4	Student Feed Back	Indirect

Course: BCA 504 P Java Programming Lab

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: **Java Programming Lab**	Semester : 5 CREDITS: 2
COURSECODE: BCA 504 P	COURSE TYPE Practical
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 3 (weekly)

Course: BCA 505T Microprocessor and Assembly Language

PROGRAMME: Computer Application	**COURSE TYPE:** Theory
DEGREE: BCA	**COURSE AREA/DOMAIN:** NA
COURSE: Assembly Language And Microprocessor	**CONTACT HOURS**: 4 (weekly)
SEMESTER: 5	**CORRESPONDING LABCOURSE CODE (IFANY):** BCA 505 P
CREDITS: 3	
COURSECODE: BCA 505T	**LABCOURSE NAME:** NA

Course Objective:

- To introduce students with the architecture and operation of typical microprocessors and Microcontrollers.
- To familiarize the students with the programming and interfacing of microprocessors and Microcontrollers.
- To provide strong foundation for designing real world applications using microprocessors and microcontrollers.

Course Outcomes:

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

1. **Assess and solve** basic binary math operations using the microprocessor and explain the Microprocessor's and Microcontroller's internal architecture and its operation within the area of manufacturing and performance.
2. **Apply** knowledge and demonstrate programming proficiency using the various addressing modes and data transfer instructions of the target microprocessor and microcontroller.
3. **Compare** accepted standards and guidelines to select appropriate Microprocessor (8085 & 8086) and Microcontroller to meet specified performance requirements.
4. **Analyze** assembly language programs; select appropriate assemble into machine a cross Assembler utility of a microprocessor and microcontroller.
5. **Design** electrical circuitry to the Microprocessor I/O ports in order to interface the processor to external devices.
6. **Evaluate** assembly language programs and download the machine code that will provide solutions real-world control problems.

Course Objectives and Role in Program

The objectives of this course include:

Teach principles of instruction set architecture and assembly language
Programming

Co.1:- Teach basic procedures of how a compiler translates C/C++ code to assembly Language and perform simple optimizations

Co.2:- Explore in detail a simple hardware CPU implementation that supports a small Instruction subset; introduce students to computer organization

Co.3:- Show how C/C++ constructs use hardware resources, and introduce concepts of efficiency and performance below the algorithmic level Students will translate a number of small C/C++ programs into assembly language, and learn to trace and debug at the assembly level. They will extend the simple CPU
Implementation introduced in class to support additional instructions
Co.4:- The knowledge of how C/C++ constructs are translated to execute on hardware, simple hardware operations and interrupt handling are crucial building blocks for the Operating Systems and Computer Architecture courses.

Learning Outcomes

- At the end of this course students will be able to
- Translate C/C++ code into assembly language
- Perform simple optimizations by hand
- Trace and debug at the assembly level
- Understand and extend simple CPU implementations
- Understand basic interrupt/exception handling
- Make simple performance estimates for assembly code

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA 505T Microprocessor and Assembly Language

PROGRAMME: Computer Application	DEGREE:BCA
COURSE Assembly Language And Microprocessor Lab	Semester : 5 CREDITS: 2
COURSECODE: BCA 505 P	COURSE TYPE Practical
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 3 (weekly)

Course: BCA 506 P Project Work

PROGRAMME: Computer Application	DEGREE:BCA

COURSE: PROJECT WORK	Semester : 5 CREDITS: 1
COURSECODE: BCA 506 P	COURSE TYPE Practical
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 8 (weekly)

Course Objectives

To learn languages to code front end and back end of a software

. - To initiate into the process of designing, coding and testing a software module.
- To develop a complete software module

Course Outcomes

Skill to apply Software Development Cycle to develop a software module.
Co.1:- Ability to use the techniques, skills and modern engineering tools necessary for software development.
Co.2:- Develop a software product along with its complete documentation

Course: BCA601T Theory of Computation

PROGRAMME: Computer Application	**COURSE TYPE:** Theory
DEGREE: BCA	**COURSE AREA/DOMAIN:** NA
COURSE: Theory Of Computation	**CONTACT HOURS:**4 (weekly)
SEMESTER:6	**CORRESPONDING LABCOURSE CODE** **(IFANY):** NA
CREDITS: 3	
COURSECODE: BCA 601T	**LABCOURSE NAME:** NA

Course pre-requisites:
Basic knowledge about fundamental concepts of Mathematics like Set algebra, elementary formal logic, constructing proofs, recurrence relations.

Course Objectives:
Introduction to finite automata, regular expressions and languages; push-down automata and context-free languages; selected advanced language theoretical topics; emphasis on technique.

Course Outcomes

Co.1:- Master regular languages and finite automata.

Co.2:- Master context-free languages, push-down automata, and Turing recognizable languages.

Co.3:- Be exposed to a broad overview of the theoretical foundations of computer science.

Co.4:- Be familiar with thinking analytically and intuitively for problem-solving situations in related areas of theory in computer science.

Co.5:- Review definitions and notations for sets, relations and functions.

Co.6:- Introduction to formal languages and Kleene's Theorem.

Co.7:- Mathematical formal proofs are including proof by induction and by contradiction.

Co.8:- The recursive definitions of regular languages, regular expressions and the use of

Co.9:- Regular expressions to represent regular languages.

Co.10:- Detailed knowledge and the relationship between regular expressions and finite automata.

Co.11:- Nondeterminism.

Co.12:-Minimal finite automata in terms of equivalence classes of strings and associated algorithm for finding minimal DFA.

Co.12:-Pumping lemma for proving that languages are not regular.

Co.13:-Context-free grammars and how to prove properties of context-free grammars.

Co.14:-Pushdown automata.

Co.15:-Application of the pumping lemma for CFL to demonstrate that a language is not context-free.

Co.16:-Turing machines (deterministic and non-deterministic) and Church-Turing Thesis.

Co.17:-Brief introduction to recursively enumerable languages.

Co.18:-Brief introduction to computability including the halting problem and related Problems and PCP

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA602 - System Programming

PROGRAMME: Computer Application	**Course Type:** Theory
DEGREE: BCA	**COURSE AREA/DOMAIN**: NA
COURSE: System Programming	**CONTACT HOURS:** 4 (weekly)
SEMESTER:6	**CORRESPONDING LABCOURSE CODE** **(IFANY): NA**
CREDITS: 3	
COURSECODE: BCA 602T	**LABCOURSE NAME:** NA

Course pre-requisites:
Basic knowledge of data structures, computer organization and architecture, operating system, and Programming skills.

Course Objectives

1. Learn basic concepts of operating systems and system software's.
2. Design of operating systems and system software's.
3. Learn the functioning of the principal parts of an operating system.

Course Outcomes

Co.1:-To understand the basics of system programs like editors, compiler, assembler, linker, loader, interpreter and debugger.
Co.2:-Describe the various concepts of assemblers and macro processors.
Co.3:-To understand the various phases of compiler and compare its working with assembler.
Co.4:-To understand how linker and loader create an executable program from an object module created by assembler and compiler.
Co.5:-To know various editors and debugging techniques.
Co.6:-To know about assembly language.
Co.7:-To know about Hypothetical machine structure.

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct

4	Student Feedback	Indirect

Course: BCA603 - Cryptography

PROGRAMME: Computer Application	**Course Type:** Theory
DEGREE: BCA	**COURSE AREA/DOMAIN**: NA
COURSE: Cryptography	**CONTACT HOURS:** 4 (weekly)
SEMESTER:6	**CORRESPONDING LABCOURSE CODE** **(IFANY): NA**
CREDITS: 3	
COURSECODE: BCA 603T	**LABCOURSE NAME:** NA

COURSE OBJECTIVES

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

COURSE OUTCOMES:

After successful completion of the course, the learners would be able to

Co.1:- Provide security of the data over the network.

Co.2:- Do research in the emerging areas of cryptography and network security.

Co.3:-Implement various networking protocols.

Co.4:-Protect any network from the threats in the world.

Assessment Methodologies

S.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct

2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA604 - Web Programming

PROGRAMME: Computer Application	**Course Type:** Theory
DEGREE: BCA	**COURSE AREA/DOMAIN**: NA
COURSE: Web Programming	**CONTACT HOURS:** 4 (weekly)
SEMESTER:6	**CORRESPONDING LABCOURSE CODE** **(IFANY): NA**
CREDITS: 3	
COURSECODE: BCA 603T	**LABCOURSE NAME:** NA

Course Objective:

- Demonstrate competency in the use of common HTML code.
- Demonstrate competency using FTP to transfer web pages to a server.
- Construct pages that meet guidelines for efficient download.
- Construct pages that meet the needs of an identified audience.
- Construct efficient file structure for web sites.
- Demonstrate proficiency in the use of WYSIWYG design software.
- Evaluate the functions of specific types of web pages in relationship to an entire web site.
- Design electronic text and web pages that include the standard textual components needed on web pages.
- Create web pages that meet accessibility needs of those with physical disabilities.
- Understand how CSS will affect web page creation.
- Understand the role of JavaScript in web page creation.
- Modify CSS and JavaScript for use on a web site.
- Understand the function of copyright in relationship to web design and coding.
- Utilize graphic design to enhance web pages.

Course Outcomes:

Each course outcome is followed in parentheses by the Program Outcome to which it relates.

Programming Environment - Install and configure an Apache 2 server with PHP5 module, MySQL database and the tool PhpMyAdmin. (*BS-CS I currency,*)

Server-side - Write a large array of programs in PHP with some of the most important functionalities the language provides. (*BS-CS I currency,*)

MySQL - Use a MySQL database with PHP to create database applications. (*BS- CS I currency,*)

Client-side - Write HTML pages and use basic JavaScript code to enhance the pages. (*BS- CS I currency*)

Advanced Client-side - Learn and use DHTML and AJAX. Learn the basics of JQuery. (*BS- CS I currency,*)

Security - Learn about the major vulnerabilities facing web sites and some simple ways to reduce their likelihood. (*BS-CS E responsibility,*)

Project - Write a complete market-ready database-driven website with PHP and JavaScript and go through the basic phases of the software life cycle (like project proposal, non- technical presentation (CEO), technical presentation (final presentation)). The project is done in groups of at least 2. (*BS-CS F communication, MS-MEM A development,*)

Assessment Methodologies

Sl.NO	DESCRIPTION	TYPE
1	Student Assignment	Direct
2	Tests	Direct
3	University Examination	Direct
4	Student Feedback	Indirect

Course: BCA 605 P Project Work

PROGRAMME: Computer Application	DEGREE:BCA
COURSE: PROJECT WORK	Semester : 6 CREDITS: 1
COURSECODE: BCA **605 P**	COURSE TYPE Practical
COURSE AREA/DOMAIN:NA	CONTACT HOURS: 8 (weekly)

Course Objectives

To learn languages to code front end and back end of a software

. - To initiate into the process of designing, coding and testing a software module.
- To develop a complete software module

Course Outcomes

Skill to apply Software Development Cycle to develop a software module.
Co.1:- Ability to use the techniques, skills and modern engineering tools necessary for software development.
Co.2:-Develop a software product along with its complete documentation

Program Specific Outcomes (PSO)

After the completion of the course, a student is able

- To pursue further studies to get specialization in Computer Science and Applications, , Mathematics
- To pursue the career in corporate sector can opt for MBA.
- To Work in the IT sector as programmer, system engineer, software tester, junior programmer, web developer, system administrator, software developer etc.
- To work in public sector undertakings and Government organizations.
- For teaching in Schools and Colleges.

Programme Outcomes (PO) – BCA

PO1: Acquire and Apply Knowledge: Ability to understand and apply the fundamental principles, concepts and methods in key areas of Computer Applications and multidisciplinary fields.
PO2: Problem Analysis: Ability to analyze real-time problems using various tools and techniques.
PO3: Design and Development: Ability to design and develop solutions to meet the desired needs.
PO4: State-of-art Technologies: Ability to adapt and apply emerging tools and technologies.
PO5: Entrepreneurship and Innovation: Ability to provide sustainable and innovative solutions for real-time problems.
PO6: Lifelong Learning: Ability to engage in continuous reflective learning in the context of technological advancement.
PO7: Communication and Team Building: Ability to demonstrate effective communication and interpersonal skills.
PO8: Ethics and Social Responsibility: Ability to integrate ethical and human values to become a socially responsible citizen.
PO9: To provide thorough understanding of nature, scope and application of computer and computer languages
P1O: To develop interdisciplinary approach among the students