COURSE OUTCOME OF COMPUTER SCIENCE

I-B.SC I- SEMESTER

Subject: Programming in C Code: BS106

Credits: Four theory hours per week with 4 credits Two practical hours per week with 1 credit

Course Objectives:

- To understand the basic concepts of the C programming language
- To analyze and evaluate the algorithm into program code with solution
- To write C program using arrays, pointers, structures and files

Course Outcomes:

CO1. Apply C Language basic concepts in constructing simple programs

CO2. Be able to write the program, edit, compile, debug, correct, recompile and run in C

CO3. To design an algorithm for the given problem

CO4. To write a C program for a given algorithm

CO5. Construct top down structured c program using functions

CO6. Explain the pointers concept for dynamic memory management and data structures implementation

CO7. Explain file handling concept for input and output processing

CO8. Ability to explain the run-time behavior of the program at machine level

CO9. Ability to develop and execute C programs.

Strategies:

- Give the significance of the subject in the field of research and its applications in other areas
- Get the students excited about programming by giving real word examples
- Provide them with small sample examples for every concept and visualize the flow of execution.

COURSE OUTCOMES(COs) C - LAB

CO 1:Understands the functional hierarchical code organization.

CO 2:Ability to handle possible errors during program execution.

CO 3:Identify solutions to a problem and apply control structures and user defined functions for solving the problem.

CO 4:Demonstrate the use of Strings and string handling functions.

CO 5:Demonstrate the different File operations for storing and handling data.

I-B.SC I- SEMESTER

Subject: (AECC) Fundamentals of computers Code: BS107 Credits: Two

theory hours per week with 2 credits

Course Objectives:

- To understand the basic concepts of Computers
- To understand Computer memory , number system and computer software

Course Outcomes:

CO1. Apply basics of computers in daily life

CO2. Be able to understand classification of computers and parts of computers

CO3. To understand what computer memory is like and various types of memory

CO4. To understand various number systems their representations CO5. Work with Boolean expressions and Boolean algebra and their

representations

CO6. Understand various types of computer software and acquiring them

Strategies:

• Give the significance of the subject in the field of research and its applications in other areas

• Provide them with small sample real time examples for every concept

I B.SC II- SEMESTER

Subject: Programming in C++ Code: BS206

Credits: Four theory hours per week with 4 credits Two practical hours per week with 1 credit

Course Objectives:

- To understand object-oriented programming concepts in developing solutions
- To learn features of C++ language including templates and exceptions Understand and learn to build C++ classes and design principles
- Understand the advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.

Course Outcomes:

- Apply C++ Language basic concepts in constructing simple programs
- Relate Object and class concept to real world entities
- Explain and write programs for Object Oriented concepts such as encapsulation, inheritance and polymorphism
- Explain file handling concept for input and output processing
- Ability to write programs for exception handling in c++
- Explain the pointers concept for dynamic memory management and data structures implementation
- Develop and Execute c++ programs in the lab for every concept.
- Ability to design and implement programs for real time complex problems

Strategies:

- Give the significance of the subject in the field of research and its applications in other areas
- Get the students excited about programming by giving real word examples
- Provide them with sample examples for every concept and visualize the flow of execution.

Course Outcomes (CO) – C++ LAB

CO 1: Implement the concepts of object oriented programming

CO 2: Implement Constructors

CO 3: Demonstrate virtual functions and inheritance

CO 4: Implement operator overloading using classes and objects

II- B.SC III- SEMESTER

Subject: Data Structures with C++ Code: BS306
Credits: Four theory hours per week with 4 credits
Two practical hours per week with 1 credit

Course Objectives:

• To introduce the fundamental concepts of data structures, abstract data types and analysis of algorithms

• To understand the implementation of linear data structures such as stacks, queues, linked lists and their applications

• To understand the implementation of different non linear data structures such as trees and graphs

• To understand various search techniques such as hashing, binary search trees

• To understand and evaluate different sorting techniques and analyze their time complexities.

Course Outcomes:

CO1. Ability to explain the basic operations on linear structures such as arrays, stacks, queues and linked lists
CO2. Ability to explain the concept of non linear structures such as trees and graphs.
CO3. Ability to explain and Evaluate search techniques
CO4. Ability to explain sorting algorithms.
CO5. Analyze the efficiency of various sorting algorithms
CO6. Construct code for polynomial expression evaluation
CO7. Develop and execute code for various sorting and search techniques.
CO8. Be able to understand hierarchical representation of data using Trees and their application in solving a wide range of problems

Strategies:

• Give the significance of the subject in the field of research and its applications in other areas

• Get the students excited about various techniques and algorithms for data representation in programming by giving real word examples

Course Outcomes - Data Structures with C++ LAB

- Analyze the performance of Trees, Graphs, Searching and Sorting techniques.
- Analyze the performance of - Stack, Queue, and Lists.
- Implement all the applications of Data structures in a high-level language.
- Design and apply appropriate data structures for solving computing problems

II-B.SC IV-SEMESTER

Subject: Database Management System BS406

Credits:Four theory hours per week with 4 credits

Two practical hours per week with 1 credit

Course Objectives:

- To understand the different issues involved in the design and implementation of a database system.

- To study the physical and logical database designs, database modeling, relational, hierarchical, and network models

- To understand and use data manipulation language to query, update, and manage a database

- To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server, Data Warehousing.

Course Outcomes:

- Differentiate database systems from file systems by enumerating the features provided by database systems

- Able to understand various data models and differentiate changes happened from one to the other model.

- Use an SQL interface of a multi-user relational DBMS package to create, secure, populate, maintain, and query a database.

- Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database.

- Get more popularize with single user databases and distributed databases.

- Be able to get familiar with what is a Business Intelligence and various methods of developing business models using OLAP, ROLAP and MOLAP.

- Be familiar with the basic issues of transaction processing and concurrency control.

- Get acquainted with data administration both at technical level and managerial level.

Strategies:

- Get the students understand basic structure of a database and create it
- Provide them with small sample examples for every concept and build their own real world examples of every concept.

Course Outcomes - DBMS LAB

- To demonstrate the use of constraints and relational algebra operations. Describe the basics of SQL and construct queries using SQL.
- Understand various advanced queries execution such as relational constraints, joins, set operations, aggregate functions, triggers and views.
- Populate and query a database using SQL DML/DDL/DCL commands.
- Programming PL/SQL including stored procedures, functions, cursors and packages.

III - B.Sc. V-SEMESTER

Subject: Python-1 Code: BS502

Credits: Two theory hours per week with 2 credits

Course Objectives:

- The course is designed to provide Basic knowledge of Python
- Develop a basic understanding of programming and the Python programming language.
- Solve problems requiring the writing of well-documented programs in the Python language, including use of the logical constructs of that language
- Be comfortable with the fact that there is more than one right solution to a problem.

Course Outcomes:

CO 1. Master an understanding of scripting and the contributions of scripting languages.

CO 2. Master an understanding of Python especially the object oriented concepts, classes

, subclasses, inheritance, and overriding.

CO 3. Understand the basics of OO design.

CO 4. Master an understanding of the built in objects of Python,
CO 5 Be fluent in the use of procedural statements assignments, conditional statements, loops, function calls and sequences.
CO 6. Be able to design, code, and test small Python programs that meet requirements expressed in English.

Strategies:

- Get the students excited about programming by giving real word examples
- Provide them with sample examples for every concept and visualize the flow of execution.

III B.Sc. V-SEMESTER

Subject: Programming in JAVA Code: BS505

Credits: Three theory hours per week with 3 credits

Two practical hours per week with 1 credit

Course Objectives:

- Understand fundamentals of object - oriented programming in Java, including defining classes, invoking methods, using class libraries, etc
- Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods etc.
- Understand the principles of inheritance, packages and interfaces.

Course Outcomes:

CO 1. Write, compile, and execute Java programs that may include basic data types and control flow constructs using J2SE or other Integrated Development Environments (IDEs).

CO 2. Knowledge of the structure and model of the Java programming language.

CO 3. Understand the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc and exception handling mechanisms. CO 4. Understand the principles of inheritance, packages and interfaces. CO 5. Demonstrate the concepts of polymorphism and inheritance.

CO 6. Write applet programs that can load images and play sound.

Strategies:

- Give the significance of the subject in the field of research and its applications in other areas
- Get the students excited about programming by giving real word examples
- Provide them with sample examples for every concept and visualize the flow of execution.

Course Outcomes (CO) – JAVA LAB

CO 1: Understand the basics of Java programming, concepts of classes, objects, inheritance and constructors.

CO 2: Implement the concepts of overloading, multi-threaded programs and Exception handling.

CO 3: To create a file, write the data and display the data.

CO 4: Analyze and use swings and AWT components and layouts in creating applications.

III B.Sc. V- SEMESTER

Subject: Elective B-Software Engineering Code: BS506

Credits: Three theory hours per week with 3 credits

Two practical hours per week with 1 credit

Course Objectives:

- The course aims to give the student insight into software engineering methods and processes.
- To understand detailed knowledge of techniques for the analysis and design of complex software intensive system
- Student uses the knowledge and information gained in the course to expand and improve the application of systems engineering in their field

Course Outcomes:

CO 1. Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle

CO 2. Design applicable solutions in one or more application domains using software engineering approaches that integrate ethical, social, legal and economic concerns.
CO 3. Ability to design different architectural styles, analyse them and thereby design models.
CO 4. Model the structure and behaviour a software system the UML class diagrams and state diagrams.
CO 5. Formulate a testing strategy for a software system, employing techniques such as system testing, software testing, object- oriented testing and black-box testing.
CO 6. Development: Graduates can effectively apply software engineering practice over the entire system lifecycle. This includes requirements engineering, analysis, prototyping, design, implementation, testing, maintenance activities and management of risks involved in software and embedded systems.

Strategies:

- Learning software engineering as multifaceted and make them aware of latest developments in software engineering education.
- To exemplify how industrial relevance can be achieved on the graduate level.
- Verify the accomplishment of goals such as preparing UML diagrams and Dataflow diagrams

Course Outcomes - Software Engineering LAB

- Create class and object diagrams in UML.
- Develop interaction, use case, activity diagrams.
- Design component and deployment diagrams.

III - B.Sc. VI-SEMESTER

Subject: Python-2 Code: BS602

Credits: Two theory hours per week with 2 credits

To Introduce the fundamental concepts of python programming language

CO1. To learn how to design and program simple Python applications using python concepts.

CO2. To learn how to build programs using package Arrays and numpy for multidimensional arrays matrices concept in Python.

CO3. To learn how to use indexing and slicing to access data in Python programs.

CO4. To learn how to write functions and pass arguments in Python.

CO5. To learn how to use Strings and Characters in Python.

CO6. To learn how to use and build programs using lists, tuples and dictionaries in Python.

Strategies:

- Give the significance of the subject in the field of research and its applications in other areas
- Provide them with sample examples for every concept and visualize the flow of execution

III- B.SC SEMESTER -VI

Subject: Computer Networking Code: BS605
Credits: Three theory hours per week with 3 credits Two practical hours per week with 1 credit

Course Objectives:

- Learn basic concepts of computer networks
- Familiarise with basic network models
- Build interest over advanced topics of computer networking and motivate them to take their masters in this discipline
- Make the student familiarise about various networking and internetworking devices they come across.

Course Outcomes:

CO1. Explain OSI/ISO model and TCP/IP models.

CO2: Understand and explain Data Communications System and its components.

CO3: Explain the types of transmission media with real time applications

CO4: Understand different types of networks in detail

CO5: Identify the different types of network devices and their functions within a network

Strategies:

- Give the significance of the subject in the field of research and its applications in other areas
- Get the students excited about various techniques of networking

Course Outcomes of Computer Networks Lab:

CO1: Understand the data transfer among all the layers of ISO-OSI model

CO2: Understand working of protocols used in network communication

CO3: Implement various system calls and identify the MAC addresses

CO4: Understand and implement concepts related to Client -Server programming.

III- B.SC SEMESTER -VI

Subject: Web Programming Code: BS606

Credits: Three theory hours per week with 3 credits Two practical hours per week with 1 credit

Course Objective:

Understand the importance of web as a medium of communication, understanding principles of creating an effective web page and learning the language of the web: HTML, CSS and java script.

Course Outcomes:

CO1. Apply a structured approach to identify needs, interests, and

functionality of a website

CO2. Different ways to select and style HTML elements using CSS

CO3. Different ways to select and style HTML elements using CSS

CO4. Students are able to develop a dynamic web page by the use of java script and DHTML

CO5. HTML tags and how to use them to start building your web pages

CO6. Students will be able to write a well formed/valid XML documents

CO7. Create an active server application using personal web server

CO8. Knowledge of various networking protocols

Strategies:

- Give the significance of the subject in the field of research and its applications in other areas
- Get the students excited about programming by giving real word examples
- Provide them with sample examples for every concept and visualize the flow of execution

Course Outcomes (CO) – Web technologies LAB

CO 1: Understand the important HTML tags for designing pages and separate design from content.

CO 2: Design web pages using CSS styles, internal and external style sheets.

CO 3: Create forms and validating forms using regular expressions.

CO 4: Develop interactive web applications through coding using HTML, CSS and Javascript.

CO 5: Design and develop a Web site using text, images, links, lists, and tables for navigation and layout

****