PES IAMS
Phone: 8147053084
8147053085
email : principaliams@pestrust.edu.in
: pesiams@pestrust.edu.in
Website : pestrust.edu.in/pesiams

ಪಿಇಎಸ್ ಇನ್‌ಸ್ಟಿಟ್ಯೂಟ್ ಆಫ್ ಅಡ್ವಾನ್ಸ್ಡ್ ಮ್ಯಾನೇಜ್‌ಮೆಂಟ್ ಸ್ಟಡೀಸ್
(ಕುವೆಂಪು ವಿಶ್ವವಿದ್ಯಾನಿಲಯದ ಸಂಯೋಜನೆಗೊಳಪಟ್ಟಿದೆ ಮತ್ತು ಕರ್ನಾಟಕ ಸರ್ಕಾರದ ಮಾನ್ಯತೆ ಪಡೆದಿದೆ)
ಎನ್ ಹೆಚ್ 206, ಸಾಗರ ರಸ್ತೆ, ಶಿವಮೊಗ್ಗ – 577 204 (ಕರ್ನಾಟಕ)
PES Institute of Advanced Management Studies
(Affiliated to Kuvempu University, Recognized by Govt. of Karnataka)
N H-206, Sagar Road, Shivamogga - 577 204 (Karnataka)

Department of Computer Science

Computers Fundamentals and C Programming

Subject Code: BSC-1

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to Computer Systems** Definition of a Computer, History of Computers, Generations of Computers, types of computer – based on size and working principle, Block diagram of a Computer with functional units(explanation), Parts of a computer system, Information processing Cycle. Definition of software and hardware, types of programming languages, assembler, compiler, interpreter, linker, loader (Definitions only),number system – decimal, binary, octal and hexadecimal number, inter-conversion of decimal to binary and vice-versa. ASCII codes. Algorithm definition, Characteristics, notations. Flowchart-definition, Symbols used in writing the flowchart Writing an algorithm and flow-chart of simple problems	10 Hours
II	**Introduction to Computer Systems** Introduction to C, features ofC , basic C program structure, character set, tokens, keywords and identifiers. Constants, variables, data types, variable declaration, symbolic constant definition.	10Hours
III	**Operators and Expressions** C operators- arithmetic, relational, logical, bitwise, assignment, increment and decrement, conditional (?:) and special operators, Arithmetic expressions, precedence of operators and associatively. Type conversions, mathematical functions.Definition of macro and pre-processor directives, Managing I/O operation – reading and writing a character, formatted and unformatted/O functions	08Hours
IV	**Control Structures-Conditional control statements-** if, if-else,nested-if,switch , go to statement, while, do-while and for statements. Unconditional control statements- break, continue and return statements(definition and explanation with syntax, flowchart and examples)	10Hours
V	**Arrays, Strings and Functions** Definitions of an array, types-one and two dimensional array,(definition, declaration, initialization with examples).Strings–definition, declaration and initialization of string variable, string handling functions- strcmp,strcpy, strrev, strlen, strlwr, strupr(explanation with syntax and examples) Functions – definition, need, syntax for function declaration, function prototype, category of functions, nesting of functions, function with arrays, scope of variables ,parameter passing mechanism call by value and call by reference. Recursion and Recursive function (definitions only)	10 Hours

Statements of Course Outcomes (COs):

By the end of the course, the student will be able to

CO-1	Demonstrate the components of Computer systems.
CO-2	Understand the basic concepts of C-Language.
CO-3	Explain different types of operators and manage I/O Operations.
CO-4	Understand the basic concepts of Decision making, branching and looping. Practical exposure to use a programming language.
CO-5	Develop practical knowledge on arrays and different types of string functions.

Department of Computer Science

DATA STRUCUTRES USING C

Subject Code: BSC-2

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to Data Structure** Definition of Structure, syntax and example for structure declaration.Definition of union, syntax and example for union declaration, difference between structure and union. Pointers–Definition, Declaration, Examples. Dynamic memory allocation functions – syntax and examples. Definition of Data Structure and types of data structures with examples.	10 Hours
II	**Stack and recursion** Definition and example of stack (LIFO), operations of stack with algorithms, applications of stack, algorithm for the conversion of infix to postfix expression. evaluation of postfix expression , Tower of Hanoi problem and factorial of a number using recursion.	10Hours
III	**Queue** Definition and example of Queue (FIFO), operations on queue, types of queue – ordinary queue and circular queue (definitions only), disadvantages of ordinary queue. Linked list–Definitions and types of lists (definitions only), operations of Single Linked List, implementation of stack using linked list, implementation of queue using linked list,	10Hours
IV	**Tree** Definition of a Tree, Definition of root, left sub tree, right sub tree, degree of node, terminal node, depth, Definition of Binary tree, types of binary trees (definition only), Algorithm for tree traversal.	10Hours
V	**Sorting and searching** Definition of sorting, explanation of bubble sort, shell sort, radix sort and merge sort with examples. Definition of searching, explanation of Binary search and linear search with examples and algorithms.	8 Hours

Statements of Course Outcomes (COs):
By the end of the course, the student will be able to

CO-1	Understand the basic concepts of pointers and structures. Practical exposure through Implementation.
CO-2	Describe stacks, evaluate expressions and understand process of recursion.
CO-3	Use different types of queues and explain its operations.
CO-4	Memorize types of binary tree and traverse them.
CO-5	Understand concepts about searching and sorting techniques.

Department of Computer Science

OBJECT ORIENTED PROGRAMMING WITH C++

Subject Code: BSC-3

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to OOPS** Object Oriented Programming paradigm, Basic concepts of Object-Oriented Programming Classes, Objects, Data Abstraction and Encapsulation, Polymorphism, Inheritance, Dynamic Binding, Message passing, Benefits of OOP, applications of OOP.	**10 Hours**
II	**Introduction to C++** Difference between C and C++, Structure of a C++ program, input and output statements, tokens - Keywords, identifiers, constants, strings and operators, reference variables – definition and example, special operators in C++, brief introduction to control structures in C++.	**10 Hours**
III	**Classes Objects and Member Functions** Difference between structure and class, syntax and example for class declaration, Definition of data member and member function, Defining member function inside and outside the class, inline functions, array of objects, default arguments, static data members and static member functions, function overloading, definition of friend function, syntax and example for the declaration of friend function, special characteristics of friend function.	**10 Hours**
IV	**Constructors, destructors and Operator overloading** Definition of a constructor, types - parameterized constructor, default constructor, copy constructor, special characteristics of constructor, definition of a destructor, special characteristics of destructor, definition to Operator overloading, overloading binary operator (+) to add two complex numbers, rules for operator overloading.	**09 Hours**
V	**Inheritance and templates** Definition of Inheritance, forms of inheritance, syntax and examples for defining derived classes, visibility modes, explanation of multilevel inheritance and hybrid inheritance with examples. Definition of templates, syntax and examples for class and function template.	**09 Hours**

Statements of Course Outcomes (COs):
By the end of the course, the student will be able to

CO-1	Understand the fundamental Object-oriented programming concepts.
CO-2	Understand the methodologies which are essential to build a good C++ program.
CO-3	Develop programs using OOPS techniques and demonstrate virtual functions and dynamic binding with polymorphism.
CO-4	Categorize various types of Constructors and implement them.
CO-5	Analyze advantages of overloading and learn how inheritance promote code reusability in C++.

Department of Computer Science

DATABASE MANAGEMENT SYSTEM

Subject Code: BSC-4

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to DBMS** Meaning of data and information, definitions of database, applications of database system, definition of DBMS, disadvantages of file processing system (advantages of DBMS), three levels of data abstraction, difference between schema and instance, definition of data models, types of data models (brief explanation), database languages – DDL and DML.	**10 Hours**
II	**E-R model** Different types of database users, functions of Database Administrator (DBA), basic-concepts - Primary keys, foreign key, super key, definition of E-R diagram, symbols used in E-R Diagram, E-R diagram for Banking enterprise, E-R diagram for Book store, types of entities, entity sets, attributes, types of attributes, weak entity sets, cardinality ratios (mapping cardinality).	**10Hours**
III	**Relational Model** Fundamental operations of Relational algebra - select, project, union, set difference, join, division operations (explanation with examples). Types of aggregate functions – MAX, MIN, SUM, COUNT and AVERAGE (Definition with example).	**10Hours**
IV	**SQL** Definition of Query, explanation of basic structure of SQL – Select, from and where clauses in SQL, data types in SQL, explanation of set operation in SQL – Union, intersection, except, NULL values.	**09Hours**
V	**Relational database design** Pitfalls in relational database design, definition of Normalization, Various types of Normal forms (Definitions only) – First Normal form, Second Normal form, Third Normal form, Boyce-Codd Normal Form (BCNF).	**09 Hours**

Statements of Course Outcomes (COs):
By the end of the course, the student will be able to

CO-1	Evaluate the role of database management systems in information technology applications within organizations.
CO-2	Develop an Entity-Relationship model based on user requirements.
CO-3	Understand the basics of relational data models. Transform Entity-Relationship diagrams to Relational Schema.
CO-4	Experience how to use SQL language for constructing and utilizing database applications.
CO-5	Interpret relational database designs and understand different types of normal forms

Department of Computer Science

JAVA PROGRAMMING

Subject Code: BSC-5.1

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to Java** History of Java, Java features, Difference between C/C++ and Java, Java program structure, Java tokens, Statements, JVM, Java and Internet, Java and WWW, Web browsers, Java support system, Java Development Kit (JDK), Application Programming Interface(API), Java Runtime Environment (JRE).Introduction to packages in Java, Applets, Operators & Expressions, Data types, Constants and Variables, Type conversions, Mathematical functions; Control Statements: Decision making and Branching with while, do-while, for and labeled loops; Arrays, Vectors & Strings: Initialization, Declaration	12 Hours
II	**Overview** Class, Objects, Constructor, Method overloading, Static members; Inheritance: Single, Multilevel, Hierarchical, Visibility modes, Method overriding, Final variable, Abstract methods and classes; Interface: Defining, Extending and implementing assigning interface variables	10Hours
III	**Packages and multithreading** Java API Packages, using system packages, naming convention, accessing and using a package, adding a class to packages, hiding classes. Multithreaded programming: Creating a thread, extending the thread class, stopping and blocking a thread, life cycle of a thread, using thread methods, thread exceptions, thread priority, synchronization, implementing the runnable interface.	10Hours
IV	**Exceptions and Debugging** Meaning of errors and exceptions, Dealing with errors, Classifications of exceptions, syntax of handling exceptions, advertising the exceptions, throwing and re-throwing exceptions, creating Exception classes, multiple catch statements, finally clause, Debugging techniques – tricks for debugging, Assertions, Java Debugger (JDB).	08Hours
V	**Applets and Graphics** Applets basics, applets and application, Life cycle, Life cycle of Applet programming-passing parameter to applets, paint and repaint methods, Graphics class, Line, Rectangle, Circle, Ellipse, Arcs and Polygon, drawing bar charts.	08 Hours

Statements of Course Outcomes (COs):
By the end of the course, the student will be able to

CO-1	Acquire basic knowledge of the structure and model of the Java programming language
CO-2	Understand Inheritance and its applications.
CO-3	Explore the principles of packages and multithreading.
CO-4	Identify exceptions and develop programs using debugging techniques.
CO-5	Develop interactive programs using applets and graphics.

Department of Computer Science

UNIX PROGRAMMING

Subject Code: BSC-5.2

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to Operating system** Definition of OS, functions of operating systems. Early systems – Simple monitors, Batch Systems, Multiprogramming, Time Sharing, Real time, Parallel and Distributed systems Scheduling concepts, Scheduling algorithms: FCFS, Shortest job first, priority scheduling, round robin, Definition of deadlock problem, deadlock characteristics, deadlock prevention and avoidance. File concept –allocation and access methods, directory structures, Contiguous allocation.	**12 Hours**
II	**Introduction to Unix** The Unix operating system, , A brief Session, The Unix Architecture, Features of UNIX, POSIX and Single UNIX specification, Locating commands, Internal and External commands, Command Structure, Flexibility of command Usage, Man Browsing the Manual Pages ON-line, Understanding the man Documentation. General-Purpose Utilities: Cal command, date command, echo, printf, bc, script, passwd, who, uname	**10Hours**
III	**The File System in Unix** The file, The Parent –Child Relationship, The HOME Variable, pwd, cd, mkdir, rmdir, Absolute Pathname, Relative Pathname, ls, The Unix File system. Handling Ordinary Files: Cat, cp, rm, mv, more, Thelp subsystem: Printing a File, File, wc, od, cmp, comm, diff, dos2unix and unix2dos, compressing and archiving files, gzip, and gunzip, tar, zip and unzip. Basic File Attributes: Listing file attributes, listing directory attributes, File Ownership, File Permissions, changing file permissions, Directory Permissions, Changing File Ownership	**10Hours**
IV	**The Vi Editor** Vi basics, Input Mode, Saving Text and Quitting, Navigation, Editing Text, Undoing Last Editing Instructions(U and U), Repeating the last command(.), Searching for a Pattern(/ and ?), Substitution	**08Hours**
V	**The Shell** The shell's Interpretive Cycle, Shell Offering, Pattern Matching, Escaping and Quoting, Redirection, /dev/null and /dev/tty, Pipes, tee, Command Substitution, Shell variables. Essential shell programming: Shell scripts, read, using command line arguments, exit and exit status of command, the logical operators && and ||- conditional execution, the if conditional, using test and to evaluate expressions, the case conditional, expr, $0: calling a script by different names, while, for, set and shift, the here document trap, debugging shell scripts with set –x, sample validation and data entry scripts	**08 Hours**

Statements of Course Outcomes (COs):

By the end of the course, the student will be able to

CO-1	Describe the architecture and features of UNIX Operating System and distinguish it from other Operating Systems
CO-2	Describe features of UNIX Operating System and distinguish it from other Operating Systems.
CO-3	Demonstrate UNIX commands for file handling and process control
CO-4	Formulate regular expressions and use them for pattern matching
CO-5	Analyze a given problem and apply requisite facets of SHELL programming to devise a SHELL script to solve the problem

Department of Computer Science

ADVANCED JAVA PROGRAMMING

Subject Code: BSC-6.1

Syllabus

UNIT	Content	Hours Allotted
I	**Review of Java Concepts and AWT, Graphics Programming** Review of Java Concepts .AWT and AWT Classes, Window fundamentals –Component, Container, Panel, Window, Frame, Canvas.Working with frame window. GraphicsProgramming: Graphics class, methods, drawing objects, line graphs, polygon classes,working with colours and fonts. Advanced graphics operations using Java2D.Designing,simple User Interfaces (UIs) using AWT, Layout Manages.	**12 Hours**
II	**Swings and event handling** Event Handling: Basics of Event Handling, the delegation event model, AWT event hierarchy and event classes, Event Listener Interfaces, Adapter Classes, Event queue. Swing: Meaning, need, difference between AWT and swing. The Model-View-Controller (MVC) designpatterns, Creating simple UIs using swing, and handling basic events.	**10 Hours**
III	**Java Beans, Java Archives (JAR)** Meaning and need of Java Beans, Advantages, Bean writing process, Bean properties. Java Archives (JARs): Meaning, need, the JAR utility, Creating JAR files.	**08 Hours**
IV	**File Management and JDBC** File, creating a file, writing to a file, opening a file, reading from a file, file management, checking existence of a file, deleting a file.JDBC: Meaning, need, concept and structure of JDBC, relation with ODBC, JDBC driver types and their meaning, the JDBC process – loading the driver, connecting to the DBMS, creating and executing SQL statement, Connection object, Statement object, Prepared Statement object, Callable Statement, Result Set, JDBC Exceptions.	**10 Hours**
V	**Basic concepts of Collections, Generics and Network programming** Collections: Meaning, need, Collection interfaces, Concrete Collections – Array List, Hash set, Map. Generics: Meaning, need, benefits, generics usage, basics of generic types, type parameter naming conventions, type wildcards, using type wildcards, generic methods, bound types, writing simple generic container, implementing the container, implementing constructors, implementing generic methods.	**10 Hours**

Statements of Course Outcomes (COs):

By the end of the course, the student will be able to

CO-1	Illustrate a full set of UI widgets and other components, including windows, menus, buttons, checkboxes, text fields, scrollbars and scrolling lists, using Abstract Windowing Toolkit
CO-2	Summarize event handling on AWT and learn how to design stand-alone desktop-oriented Graphical User Interface based Java applications using Swing/Java Foundation Components.
CO-3	Develop and deploy enterprise beans using the Java EE architecture
CO-4	Understand basic file operations, file management, and learn to access database through Java programs, using Java Data Base Connectivity
CO-5	Compare the types of interfaces in Collections Framework and understand the objective of generic programming and implement generic classes and methods.

Department of Computer Science

SOFTWARE ENGINEERING AND COMPUTER NETWORKS

Subject Code: BSC-6.2

Syllabus

UNIT	Content	Hours Allotted
I	**Introduction to Software Engineering** IEEE definition of Software and Software Engineering, Software Problems, Software engineering challenges, Software quality attributes, phases in software development (Phased Development process), Definition of Software process, Components of software process, desired characteristics of software process, Software development process models- waterfall model, prototype model and spiral model	12 Hours
II	**Software design** Definition of SRS, need for SRS, Characteristics of SRS, Structure of SRS, design objectives, design principles, module level concepts – coupling and cohesion	10 Hours
III	**Coding and testing** Definition of Coding, Programming principles and guidelines, top down and bottom-up Approaches, definition of testing, testing fundamentals, levels of testing, Difference between black box testing and white box testing.	08 Hours
IV	**Introduction to Computer networks Network Hardware** Definition of computer network, Goals of computer network, Types of Networks based on transmission technology - Broadcast, point- to -point, Types of Networks based on size & scale - LAN, WAN, MAN, Protocol hierarchies (Network software), Network topologies – Bus, Mesh, Ring, tree and star	10 Hours
V	**Network Software, Reference models and Transmission Media** Reference models - OSI / ISO model, TCP / IP model, ARPANET, Transmission Media - twisted pair, coaxial cable, fiber optics cable, Internet and its applications, Wireless media - Bluetooth, Wi-Fi, internet and its applications	10 Hours

Statements of Course Outcomes (COs):

By the end of the course, the student will be able to

CO-1	Acquire basic knowledge and understanding of the software engineering challenges, analysis and design of complex systems
CO-2	Specify software requirements through a productive working relationship with various stakeholders of the project
CO-3	Develop the code from the design and effectively apply relevant standards and perform testing, and quality management and practice
CO-4	Conclude the usage of computer networks, network hardware and its types.
CO-5	Understand reference models, protocol & its types, Example networks-ARPANET, ATM.