SEMESTER III								
S.No.	Course Code	Course Title	Credit					
			L	T	P	C	Int.	Ext.
1	BCA113	Core Java	2	1	0	3	40	60
2	BCA114	Design & Analysis of Algorithms	2	1	0	3	40	60
3	BCA115	Introduction to System Software	2	1	0	3	40	60
4	BCA116	Principles of Programming Languages	2	1	0	3	40	60
5	BCA117	Fundamentals of Computer Networks	2	1	0	3	40	60
6	BCA118	LINUX & Shell Programming	2	1	0	3	40	60
7	BCA235	Lab I: Core Java	0	1	3	4	40	60
	BCA236	Lab II: LINUX & Shell Programming	0	1	3	4	40	60
			12	2	12	26		

SEMESTER III

Paper 1: Core JAVA (BCA113)

Unit 1: Introduction to Java and Its environment: Programming language types, Computer programming hierarchy, Why Java?,Flavors of Java, Java Designing goals, Features of Java, JVM, Bytecode, Installing Java, Java Program Development and Deployment, Java Source File Structure, Compilation (JIT, AIT), Execution, Naming conventions, Unicode systems, difference between JDK, JRE and JVM.

Unit 2: Basic Language Elements: Lexical Tokens, Identifiers, keywords, Literals, Comments, Datatypes, Operators and expressions; decision making, branching and looping.

Unit 3: Object Oriented Programming: Advantages of OOPs,Class Fundamentals, Object and Object Reference, Method Overloading, constructor, Static (Variable, Method and Block), this keyword, Inheritance (IS-A), Aggregation and Composition (HAS-A), method overriding, covariant return type, super keyword, instance initializer block, final keyword, runtime polymorphism, static and dynamic binding, abstract class and interface, down casting with instanceof operator, Access modifiers, encapsulations, object class, object cloning, Java Array, Call by value and call by reference strictfp keyword.

Unit 4: Package and Exception Handling: Organizing classes and Interfaces in packages, Package as Access Protection, Defining Package, Class path setting for packages, Making JAR files for Library packages, Import and Static import naming convention for packages. Exception and Errors, types of exception, control flow in exceptions, JVM reaction to exceptions, use of try, catch, finally, throw, throws in exception handling, in-build and user defined exception handling, checked and un-checked exceptions.

Unit 5: String and Input-Output in Java: String: what and why? Types of strings, string comparison, methods of string class, StringBuffer Class, StringBuilder Class, Creating Immutable String, toString Method, StringTokenizer class. FileOutputStream and FileInputStream, ByteArrayOutputStream, SequenceInput Stream, BufferedOutput Stream and BufferedInputStream, FileWriter and File Reader, Char Array Writer, Input from Keyboard by InputStreamReader, Input from Keyboard by console, input from keyboard by scanner, printStream class, PrintWriter class, Compressing and Uncompressing file, reading and writing data simultaneously, DataInputStream and DataOutputStream, Steam Tokenizer class.

References:

1. Burdman, "Collaborative Web Development", Addison Wesley.
2. Ivan Bayross, "Web Technologies Part II", BPB Publications.
3. Margaret Levine Young, "The Complete Reference: JAVA", TMH
4. Naughton, Schildt, "The Complete Reference JAVA2", TMH

Paper-2: Design & Analysis of Algorithms (BCA114)

Unit 1: Basics of an Algorithm: Definition and Example of an algorithm, Characteristics of an algorithm, Steps in Designing of Algorithms, Growth of function, Recurrence, Problem Formulation, Solving Recurrence Relations using Substitution Method, Iteration Method, Master Method. **Asymptotic Bounds**: Performance Analysis-Space Complexity, Time Complexity, Asymptotic Notations, Big Oh Notation O, Omega Notation Ω, Theta Notation θ and Little Oh Notation, Limiting Behavior Of Asymptotic Notations, Concept of efficiency of analysis of an algorithm, Basic Algorithms and Comparative efficiencies of algorithms: Linear, Quadratic, Polynomial and Exponential.

Unit 2: Searching: Linear Search, Binary Search and their Complexity Analysis. **Sorting:** Concept of Sorting, Selection Sort, Insertion Sort, Bubble Sort and their Complexity Analysis. **Linear Time Sorting**: Radix Sort. Counting Sort, Bucket Sort. **Divide and Conquer Approach**: Quick Sort and Merge Sort, Strassen's Matrix Multiplication.

Unit 3: Graph Algorithms and Trees: Cyclic graph, Acyclic graph, Directed Acyclic Graph, Representation of Graphs, Adjacency Matrix, Adjacency List, Depth First Search and Examples, Breadth First Search and Examples, Topological Sort. Greedy Method: **Minimum Spanning Tree:** Kruskal's Algorithms, Prim's Algorithms, **Shortest Path: Single Source**: Dijkstra, Bellman Ford, **All pair shortest Path:** Floyd's Algorithms **Dynamic Programming:** General approach, multi-stage graph, matrix-chain multiplication, all-pairs shortest paths, travelling salesperson, 0/1 knapsack problem, longest common subsequence.

Unit 4: Backtracking: N-queen problem, sum of subsets, knapsack problem, generation of all cliques, traveling salesperson problem, Graph coloring. **Branch-and-Bound:** Assignment problem, 0/1 knapsack problem. **Randomizing Algorithms:** Numerical Integration, Primality testing, randomized min-cut, randomized algorithm for n-queens, quick-sort.

Unit 5: Approximation and Lower Bound Theory: Job scheduling, Bin packing, set cover, Max cut. **Lower Bound Theory:** Decision tree; Reduction method; NP-completeness, Cook's Theorem, Examples of NP-Hard and NP-Complete problems

References:

1. Fundamental of Computer algorithms – Horowitz and Sahni
2. The art of Computer Programming – Donald Knuth
3. Design Methods and Analysis of Algorithms – S.K. Basu
4. The Design and Analysis of Computer Algorithms – Aho, Hopcraft and Ullaman
5. Genetic Algorithm in Search, Optimization and Machine Learning – David E. Goldberg

Paper 3: Introduction to System Software (BCA115)

Unit 1: General concepts-Review of assembly and machine language programming, distinction between system software and application software, Language processors:- Introduction, Language processing activities.
Assemblers:- Elements of Assembly language programming, A simple assembly scheme, Pass structure of assemblers, Design of two pass assemblers.
Unit 2: Macros and macro processors:- Macro definition and call, Macro expansion, Nested macro calls, advanced macro facilities, design of macro pre processor Linker-Relocation and linking concepts-self relocating programs. Loader-Types of loaders Editor-Types of editors-Components of editor-Debug monitor.
Unit 3: Introduction to compiling:- Compilers, Analysis of a source program, the phases of a compiler, Lexical analysis:-The role of the lexical analyzer, Input buffering, specification of tokens Recognition of tokens, Finite automata, Conversion of an NFA to DFA, From a regular expression to an NFA.
Unit 4: Syntax analysis:- the role of the parser, Context free grammars, writing a grammer, Top dowm parsing Bottom up parsing, syntax directed translation-syntax directed definition, Construction of Syntax Tree, L R parsers-LR parsing algorithm, Constructing SLR parsing tables, SLR parsing table.
Unit 5: Intermediate code generation-postfix notation, syntax tree, three-address code, basic blocks and flow graph, the DAG representation of basic blocks, Backpatching, Code optimization:- The principal sources of optimization, optimization of basic blocks, loops in flow graphs, Peephole optimization Code Generations:- Issues in the design of a code generator, simple code generator.
References:
1. Systems Programming- Donovan
2. Introduction to Systems Software- Dhamdhere D.M.

Paper 4: Principles of Programming Languages (BCA116)

Unit 1: Introduction: The Role of Programming Languages: Why Study Programming Languages, Towards Higher-Level languages, Programming paradigms, Programming environments. Language Description: Syntactic structure, language Translation Issues: Programming language Syntax, Stages in translation, Formal translation Models.

Unit 2: Language Properties: Modeling Language Properties, Elementary Data Types, Encapsulation, Inheritance, Sequence Control, Subprogram Control.

Unit 3: Programming Paradigms: Imperative Programming: Statements, Types, Procedure Activations Object-Oriented Programming: Grouping of Data and Operations, object oriented programming Functional Programming: Elements, Programming in a typed language, Programming with lists.

Unit 4: Other Programming Paradigms: Logic Programming, Concurrent Programming, Network Programming, Language Description: Semantic Methods.

Unit 5: Lambda Calculus: Introduction to Lambda Calculus, Simple types, Sub-typing

References:

1. "Programming Languages: Design and Implementations" , Terrance W.Pratt, Marvin V.Zelkowitz, T.V.Gopal, Fourthed, Prentice Hall
2. "Programming Language Design Concept", David A. Watt, Willey India
3. "Programming languages: Concepts and Constucts", Ravi Sethi, Second Ed.,Pearson.
4. "Types and programming Languages", Benjamin C. Pierce. The MIT Press Cambridge, Massachusetts London, England.

Paper 5: Fundamentals of Computer Networks (BCA117)

Unit 1: Introduction: History of data communication, Open system standard, Definition of communication link andits application in telephony and computer networks, Importance of channel bandwidth and system noise, Protocols in telephony and internet communication, Types of channel, Advantages and disadvantages of analog and digital transmissions, Digitizing Speech, Wave form coding and companding, Voice over IP.

Unit 2: Data Transmission Basics: Synchronous/Asynchronous, Error detection and correction methods, Data compression, Protocol basic, Circuit, Message, Packet and Cell switching, Connection oriented and connectionless services, importance of modulation and multiplexing in communication: introduction to different modulation and multiplexing techniques; importance of Nyquist Criterion and Shannon's theorem in communication; delay, bandwidth, throughput and noise.

Unit 3: Computer Networks: Advantages and disadvantages of computer networks; classification of computer networks; introduction to various physical media in connection oriented and connection less networks; network protocols and their role in computer network. Layered approach to network design- ISO/OSI and TCP/IP model.

Unit 4: Network Topology and Network Devices: Network topology, LAN wired/wireless, Ethernet, CSMA/CD, CSMA/ CA, Token passing rings, FDDI, Introduction to networking devices- repeaters, hubs, Switches, Bridges, Routers and gateways, Switching techniques: Store and forward, Filter, Next-Hop forwarding, Introduction to routing techniques- Link state routing and distance Vector routing.

Unit 5: Internetworking: IP addressing, Address binding with ARP, Datagram encapsulation and fragmentation, Sub-netting and implementation of CIDR, UDP and TCP, TCP segment format, Adaptive retransmission, ICMP and error handling. Network applications, Client-Server concepts and application, DNS, HTTP, Email and web browsing, Broadband Multi-Service networks, Cell based networks, ISDN.

References:

1. Computer Networks :Tanenbaum, A.S
2. Data and Computer communication :Stallings, William
3. Inter Networking With TCP/IP Vol I, II,III: Comer, D.E. and Stevens D.L.
4. Local Networks : Stalling, William
5. Data Communication and Networking : Forouzan, B.A

Paper 6: Linux and Shell Programming (BCA118)

Unit 1: Introduction To Linux And Linux Utilities: A brief history of LINUX, architecture of LINUX, features of LINUX, introduction to vi editor. Linux commands- PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, gzip, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, unlink, du, df, mount, umount, find, unmask, ulimit, ps, w, finger, arp, ftp, telnet, rlogin.Text Processing utilities and backup utilities , tail, head , sort, nl, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk, cpio.

Unit 2: Introduction to Shells and Filters: Linux Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command-Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization. Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count Characters, Words or Lines, Comparing Files.

Unit 3: File Structure and Management: Grep: Operation, grep Family, Searching for File Content. Sed: Scripts, Operation, Addresses, commands, Applications, grep and sed. UNIX FILE STRUCTURE: Introduction to UNIX file system, inode (Index Node), file descriptors, system calls and device drivers. File Management: File Structures, System Calls for File Management – create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API – opendir, readdir, closedir, mkdir, rmdir, umask.

Unit 4: Process and Signals: Process, process identifiers, process structure: process table, viewing processes, system processes, process scheduling, starting new processes: waiting for a process, zombie processes, orphan process, fork, vfork, exit, wait, waitpid, exec, signals functions, unreliable signals, interrupted system calls, kill, raise, alarm, pause, abort, system, sleep functions, signal sets. File locking: creating lock files, locking regions, use of read and write with locking, competing locks, other lock commands, deadlocks.

Unit 5: Inter Process Communication: Pipe, process pipes, the pipe call, parent and child processes, and named pipes: fifos, semaphores: semget, semop, semctl, message queues: msgget, msgsnd, msgrcv, msgctl, shared memory: shmget, shmat, shmdt, shmctl, ipc status commands. **Introduction to Sockets**: Socket, socket connections - socket attributes, socket addresses, socket, connect, bind, listen, accept, socket communications.

References:

1. Linux & Shell Programming
2. Beginning Shell Scripting by Erick Foster-Johnson, Wiley India
3. Beginning Linux Programming, Neil Mathew, Richard Stones, Wiley India

BCA235 Lab I: Core Java
BCA236 Lab II: LINUX & Shell Programming