

B.Sc./B.A.(Voc.) in Computer Applications

1st Year: Semester I

Course Objectives:

- To gain basic knowledge of digital electronics circuits and its levels.
- To understand and examine the computer architecture and networking.
- To learn about the basic requirements for a design application.
- To understand the logic functions, circuits, truth table and Boolean algebra expression.

CC-1: Computer Fundamentals and Organizations

[Credits: 4, Lectures: 60]

Unit 1. Introduction to Computer and Basic anatomy (12 Lectures)

Definitions: Data, Processing, information, Bits, Bytes and Words, Input, Output, CPU, Peripheral Devices, Backing Storage Inputinterface, Output Interface, Microprocessor, RISC, CISC, Types of motherboards Expansion slots, Adaptor Cards, SMPS, Controller Cards, Video – Raster Scanning & interleaved double scan, DISPLAYS – MDA, CGA, HGA, EGA, VGA, SVGA

Unit 2. Classification of Computers and Applications (6 Lectures)

On the basis of generation Micro, Mini, Super, Mainframe, Digital, Analog, Hybrid Applications in Business & Industry, Science & Technology, Education, Health, Communication, Banking & Other Public services, Block Diagram of IBM PC,

Unit 3. Peripheral Devices (12 Lectures)

Input Devices

Keyboard, Mouse, Joystick, Trackball, Touch Panels, OMR, OCR, MICR, Data Glove, Digitizers, Image scanners, Light Pen, Voice system

Output Devices

Printers: Impact printers and non-impact printers, Character Printers, Line Printers and Page Printers, Dot matrix, Daisy wheel, Thermal, LASER, Inkjet, Electrostatic, Drum Printer, Chain Printer

Plotters: Drum and Flat Bed Plotters

VDU: Visual Display Adapters, LCD

Secondary Storage Devices

Magnetic Disk (Winchester and hard disks),

Magnetic Tape, Floppy Disk, Optical (Video or LASER disk), CD

Unit 4. Memory and Register (9 Lectures)

Primary memory, Secondary Memory, Cache memory
RAM-SRAM, DRAM,

ROM-PROM, EPROM, EEPROM, CDROM, ROM BIOS, BUSES
Registers- Program Counter, Memory Buffer register, Memory Address
Register, Accumulator, Cache Memory

Unit 5. Computer Codes and Number system (9 Lectures)

Number System:(i) Positional – Decimal, Binary, Octal, Hexadecimal(ii) Non Positional (Roman), Conversion of Decimal Number in other systems and vice-versa
Binary Arithmetic – Addition, Subtraction (1’s complement, 2’s complement), Multiplication, Division, Boolean Algebra – BASIC Gates (OR, AND, NAND, NOT, XOR, NOR)

Unit 6. Types of processing and Introduction to Operating Systems (7 Lectures)

(i) Batch Processing(ii) On-line processing(iii) Real-time processing
Operating Systems(OS), Multi-programming, Multitasking, Multiprocessor, Utilities,
Compilers/Interpreters, Assemblers

Unit 7. Networking (5 Lectures)

Topologies(BUS, STAR, RING, MIXED), Advantages and disadvantages of networking,
LAN, WAN, Transmission Media, Multimedia, Internet

Reference Books:

1. Computer Fundamental Architecture and Organization – B. Ram.
2. Computer Fundamentals – Sukhvir Singh (Khanna Publication)
3. “O” Level (Module – I) Information Technology – V.K. Jain
4. Introduction to IBM PC – Peter Nortan
5. Microprocessor Architecture, Programming Applications – Ramesh S.Gaonakar
6. Illustrated MS-DOS 6.22 – Russell A. Stultz

Core Course 1 Lab: DOS [Credits: 2, Lab: 30]

Directory Structure, File naming Convention and valid characters for filenames.

Internal Commands:CLS, DIR, COPY CON, TYPE, REN, COPY, DATE, TIME, DEL, MD, RD, CD,ERASE, PROMPT, ECHO, PATH, PAUSE, SET, VER, VOL, REM.(.) Current Directory, (..) Root Directory, External Commands:FORMAT, CHKDSK, SCANDISK, DISKCOPY, SORT, FIND, SHIFT, ATTRIB,XCOPY, TREE, UNDELETE, EDIT, DELTREE, MOVE, DEFRAG, MORE, IF EXIST,BREAK, VERIFY,Redirection, Pipes, Command Line Parameter. Executable Files, Text Files,Batch Files.

Course Outcome:

- Skill to build and troubleshoot digital logic circuits.
- Ability to interpret logic gates and its operations.
- Familiarization with semiconductor memories in electronics.
- Familiarization with computer architecture and networking.

Course Objective:

- To introduce students to a powerful programming language – C.
- To understand the basic structure of a C program.
- To gain knowledge of various programming errors.
- To enable the students to make flowchart and design an algorithm for a given problem.
- To enable the students to develop logics and programs.

CC-2: Programming Techniques Using C

[Credits: 4, Lectures: 60]

Unit 1: Introduction to Programming (6 Lectures)

History of C, Overview of Procedural Programming, Introduction to Algorithm & Flowcharts.

Unit 2: Understanding Compilation and Execution in C (6 Lectures)

Using main() function, Compiling and Executing Simple Programs in C.

Unit 3: Data Types, Variables, Constants, Operators and Basic I/O (7 Lectures)

Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putchar), Formatted and Console I/O (printf(), scanf()), Using Basic Header Files (stdio.h, conio.hetc).

Unit 4: Expressions, Conditional Statements and Iterative Statements (7 Lectures)

Simple Expressions in C(including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

Unit 5: Understanding Functions (6 Lectures)

Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.

Unit 6: Implementation of Arrays and Strings (6 Lectures)

Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays

Unit 7: User-defined Data Types (Structures and Unions) (4 Lectures)

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

Unit 8: Pointers and References in C (8 Lectures)

Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring and initializing references, Using references as function arguments and function return values

Unit 9: Memory Allocation in C (3 Lectures)

Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, use of new and delete operators, storage of variables in static and dynamic memory allocation

Unit 10: File I/O (4 Lectures)

Opening and closing a file, Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files,

Unit 11: Preprocessor Directives (3 Lectures)

Understanding the Preprocessor Directives (#include, #define)

Reference Books:

1. ANSI C – E. Balagurusamy
2. Programming in C – Gottfried (TMH).
3. Let us C – YashwantKanetkar
4. "The C Programming Language ANSI C Version", Kernighan & Ritchie, Prentice Hall Software Series
5. "ANSI C - Made Easy", Herbert Schildt, Osborne McGraw-Hill
6. "Learning to Program in C", N. Kantaris, Babani
7. "C - The Complete Reference", Herbert Schildt, Osborne McGraw-Hill
8. "Programming in C", ReemaThareja, Oxford University Press
9. "A First Course in Programming With C", T. Jeyapoovan, Vikas Publishing House
10. "Let Us C", Yashavant P. Kanetkar, BPB Publications

Core Course 2- Lab

[Credit: 2, Lab: 30]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. WAP to perform input/output of all basic data types.
2. WAP to enter two numbers and find their sum.
3. WAP to reverse a number.

4. WAP to Swap Two Numbers.
5. WAP to Check Whether a Number is Even or Odd
6. WAP to compute the factors of a given number.
7. WAP to enter marks of five subjects and calculate total, average and percentage.
8. WAP to print the sum and product of digits of an integer.
9. WAP to Check Whether a Character is Vowel or Consonant
10. WAP to Find the Largest Number Among Three Numbers
11. WAP to compute the sum of the first n terms of the following series $S = 1 - 2 + 3 - 4 + 5 - \dots$
12. WAP to compute the sum of the first n terms of the following series $S = 1 + 1/2 + 1/3 + 1/4 + \dots$
13. WAP to print a triangle of stars as follows (take number of lines from user):

```

      *           1           1           1 A           A
     ***         11         2 3         1 2 A B         BA
    *****      121       4 5 6       1 2 3 A B C       CBA
   *****      1331      7 8 9 10     1 2 3 4 A B C D
          DCBA
 *****      14641      11 12 13 14 15  1 2 3 4 5 A B C D E
EDCBA

```

14. WAP to perform following actions on an array entered by the user:
 - a) Print the even-valued elements
 - b) Print the odd-valued elements
 - c) Calculate and print the sum and average of the elements of array
 - d) Print the maximum and minimum element of array
 - e) Remove the duplicates from the array
 - f) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

15. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
16. Write a program that swaps two numbers using pointers.
17. Write a program in which a function is passed address of two variables and then alter its contents.
18. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
19. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator.
20. Write a menu driven program to perform following operations on strings:
21. Show address of each character in string
22. Concatenate two strings without using strcat function.
23. Concatenate two strings using strcat function.
24. Compare two strings
25. Calculate length of the string (use pointers)
26. Convert all lowercase characters to uppercase

27. Convert all uppercase characters to lowercase
28. Calculate number of vowels
29. Reverse the string
30. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
31. WAP to display Fibonacci series (i)using recursion, (ii) using iteration
32. WAP to calculate Factorial of a number (i)using recursion, (ii) using iteration
33. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.
34. Write a menu-driven program to perform following Matrix operations (2-D array implementation):
 - a) Sum
 - b) Difference
 - c) Product
 - d) Transpose
35. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
36. Write a program to retrieve the student information from file created in previous question and print it in following format:

Roll No.	Name	Marks
----------	------	-------
37. Copy the contents of one text file to another file, after removing all whitespaces.
38. WAP to Write a Sentence to a File.
39. WAP to Read a Line From a File and Display it.
40. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
41. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

Course Outcome:

- In-depth understanding of various concepts of C language.
- Ability to read, understand and trace the execution of programs.
- Skill to debug a program.
- Skill to write program code in C to solve real world problems.

B.Sc./B.A.(Voc.) In Computer Applications

1st Year: Semester II

Course Objective:

- To learn the syntax and semantics to write Java programs.
- To understand the fundamentals of object-oriented programming in Java.
- To familiarize with the concept of inheritance, polymorphism, packages and interfaces.

Core Course -3: Programming in JAVA

[Credit: 4, Lecture Hours: 60]

Unit 1: Introduction to Java (4 Lectures)

Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

Unit 2: Arrays, Strings and I/O (8 Lectures)

Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

Unit 3: Object-Oriented Programming Overview (4 Lectures)

Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

Unit 4: Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata (14 lectures)

Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

Unit 5: Exception Handling, Threading, Networking and Database Connectivity (15 Lectures)

Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

Unit 6: Applets and Event Handling (15 Lectures)

Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts. Overview of servlets.

Reference Books:

1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.
2. James Gosling, Bill Joy, Guy L Steele Jr, GiladBracha, Alex Buckley"The Java Language Specification, Java SE 8 Edition (Java Series)", Published by Addison Wesley, 2014.
3. Joshua Bloch, "Effective Java" 2nd Edition,Publisher: Addison-Wesley, 2008.
4. Cay S. Horstmann, GaryCornell, "Core Java 2 Volume 1 ,9th Edition,Printice Hall.2012
5. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 2 - Advanced Features)", 9th Edition, Printice Hall.2013
6. Bruce Eckel, "Thinking in Java", 3rd Edition, PHI, 2002.
7. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.2009.
8. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.
9. Bert Bates, Kathy Sierra, "Head First Java", Orielly Media Inc. 2nd Edition, 2005.
10. Object Oriented Programming through JAVA, P Radha Krishna, University Press
11. David J. Eck, "Introduction to Programming Using Java", Published by CreateSpace Independent Publishing Platform, 2009.
12. John R. Hubbard, "Programming with JAVA", Schaum's Series, 2nd Edition, 2004.
13. R. Nageswara, Core Java: An Integrated Approach, Dreamtech Press

Core Course 3 (Programming in JAVA) Lab[Credit: 2, Lab: 30]

Students are advised to do laboratory/practical practice not limited to, but including the following types of problems:

1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. To learn use of .lenth in case of a two dimensional array
5. To convert a decimal to binary number
6. To check if a number is prime or not, by taking the number as input from the keyboard
7. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
8. Write a program that show working of different functions of String and StringBufferclasses like setCharAt(, setLength(), append(), insert(), concat()and equals().
9. Write a program to create a —distance|| class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer

10. Modify the `Distance` class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
11. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)
12. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword
13. Write a program to show the use of static functions and to pass variable length arguments in a function.
14. Write a program to demonstrate the concept of boxing and unboxing.
15. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
16. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacci series is given in a different file belonging to the same package.
17. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. Write a program `DivideByZero` that takes two numbers a and b as input, computes a/b, and invokes `ArithmeticException` to generate a message when the denominator is zero.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
20. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of `Exception` which itself is a subclass of `Throwable`).
21. Write a program to demonstrate priorities among multiple threads.
22. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
23. Write a program to create `URL` object, create a `URLConnection` using the `openConnection()` method and then use it examine the different components of the `URL` and content.
24. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
25. Write a program that creates a `Banner` and then creates a thread to scrolls the message in the banner from left to right across the applet's window.
26. Write a program to get the `URL/location` of code (i.e. java code) and document(i.e. html file).
27. Write a program to demonstrate different mouse handling events like :-
`mouseClicked()`, `mouseEntered()`, `mouseExited()`, `mousePressed`, `mouseReleased()` and `mouseDragged()`.
28. Write a program to demonstrate different keyboard handling events.
29. Write a program to generate a window without an applet window using `main()` function.
30. Write a program to demonstrate the use of push buttons.

Course Outcome:

- Skill to write Java application programs using OOP principles and proper program structuring.
- Ability to create packages and interfaces.

- Ability to implement error handling techniques using exception handling.

Course Objective:

- To enable the students to understand the functionality and implementation of computer system.
- To familiarize with the various instruction codes and formats of different CPUs.
- To introduce the students to I/O and memory organization of computer system.
- To deliver an overview of Control Unit of a computer system.
- To learn the usage of parallel and vector processing.

Core Course-4: Computer System Architecture

[Credits: 6(5 + 1), Lectures: 60]

Unit 1: Basic Computer Organization and Design (13 lectures)

Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, input-output and interrupt, Interconnection Structures, Bus Interconnection design of basic computer.

Unit 2: Central Processing Unit (15 lectures)

Register organization, arithmetic and logical micro-operations, stack organization, micro programmed control. Instruction formats, pipelining and parallel processing.

Unit 3: Memory Organization (8 lectures)

Shift registers, Multiplexers, Encoder, Decoder, Half Adder, Full Adder, Cache memory, Associative memory, mapping.

Unit 4: Processors (14 lectures)

8086, 8088, 80286, 80386, 80486, P5, Dual Core, Core2Duo, Quadra core, Octa core, Real Mode & Protected Mode, addressing modes, instruction codes, Difference between DX & SX of different microprocessor,

Unit 5: Bus Architecture (10 lectures)

Bus width, Clock Pulse Generator, DMA controller, PIC (Programmed InterruptController), memory, ports, Communications, Architecture – ISA, EISA (Brief Description)

Reference Books:

1. M. Mano, Computer System Architecture, Pearson Education 1992
2. A. J. Dos Reis, Assembly Language and Computer Architecture using C++ and JAVA, Course Technology, 2004
3. W. Stallings, Computer Organization and Architecture Designing for Performance, 8th Edition, Prentice Hall of India, 2009
4. M.M. Mano , Digital Design, Pearson Education Asia, 2013
5. Carl Hamacher, Computer Organization, Fifth edition, McGrawHill, 2012.

Course Outcome:

- Ability to understand the functionality, organization and implementation of computer system.
- Skill to recognize the instruction codes and formats.
- Knowledge of the internal working of main memory, cache memory, associativememoryand various modes of data transfer.
- Familiarization with the working of parallel processing and vector processing.

B.Sc./B.A.(Voc.) In Computer Applications

2nd Year: Semester III

CC5: Python Programming [Credit: 4, Lecture Hours: 60]

Course Objective:

This course covers the basics and advanced Python programming to harness its potential for modern computing requirements.

Unit 1: Introduction and overview

(10 Lectures)

Introduction, What is Python, Origin, Comparison, Comments, Operators, Variables and Assignment, Numbers, Strings, Lists and Tuples, Dictionaries, if Statement, while Loop, for Loop and the range() Built-in Function, Files and the open() Built-in Function, Errors and Exceptions, Functions, Classes, Modules Syntax and Style Statements and Syntax, Variable Assignment, Identifiers, Basic Style Guidelines, Memory Management, Python Application Examples

Unit 2: Python Objects

(10 Lectures)

Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions. Sequences: Strings, Lists, and Tuples, Sequences, Strings, Strings and Operators, String-only Operators, Categorizing the Standard Types, Unsupported Types. Numbers and Strings. Built-in Functions, String Built-in Methods, Special Features of Strings

Unit 3: Lists

(10 Lectures)

Operators, Built-in Functions, List Type Built-in Methods, Special Features of Lists, Tuples, Tuple Operators and Built-in Functions, Special Features of Tuples

Dictionaries

Introduction to Dictionaries, Operators, Built-in Functions, Built-in Methods, Dictionary Keys, **Conditionals and Loops:** if statement, else Statement, elif Statement, while Statement, for Statement, break Statement, continue Statement, pass Statement, else Statement

Unit 4: Files and Input/Output

(10 Lectures)

File Objects, File Built-in Function, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules

Unit 5: Advance topics (10 Lectures)

Security – Encryption and Decryption , Classical Cyphers

Graphics and GUI Programming – Drawing using Turtle, Tkinter and Python, Other GUIs

Unit 6: Errors and Exceptions(10 Lectures)

What Are Exceptions? Exceptions in Python, Detecting and Handling Exceptions, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions

Functions: Functions, Calling Functions, Creating Functions, Formal Arguments, Positional Arguments, Default Arguments, Why Default Arguments?, Default Function Object Argument Example, Variable-length Arguments, Non-keyword Variable Arguments (Tuple), Keyword Variable Arguments (Dictionary)

[Credits: 2, Lab: 30]

Software Lab Based on Python: Students are advised to do laboratory/practical practice.

Reference Books:

1. John V Guttag. “Introduction to Computation and Programming Using Python”, Prentice Hall of India
2. R. Nageswara Rao, “Core Python Programming”, dreamtech
3. Wesley J. Chun. “Core Python Programming - Second Edition”, Prentice Hall
4. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Python”, Wiley
5. Kenneth A. Lambert, “Fundamentals of Python – First Programs”, CENGAGE Publication
6. Luke Sneeringer, “Professional Python”, Wrox
7. “Hacking Secret Ciphers with Python”, Al Sweigart, URL-
<https://inventwithpython.com/hacking/chapters>

Course Outcome:

After learning the course, the student will be able:

1. To develop proficiency in creating based applications using the Python Programming Language.
2. To be able to understand the various data structures available in Python programming language and apply them in solving computational problems.
3. To be able to do testing and debugging of code written in Python.
4. To be able to draw various kinds of plots using PyLab.
5. To be able to do text filtering with regular expressions in Python
6. To be able to create socket applications in Python
7. To be able to create GUI applications in Python

CC -6: Operating Systems

[Credit: 4, Lecture Hours: 60]

Course Objective:

To study the basic concepts and functions of operating systems and learn the basics Unix Commands

Unit 1: Introduction (06 Lectures)

Basic OS functions, resource abstraction, types of operating systems—multiprogramming systems, batch systems, time sharing systems; operating systems for personal computers & workstations, process control & real time systems. Multiuser, multitasking, multiprocessing and real time operating systems,

Unit 2: Operating System Organization (6 Lectures)

Processor and user modes, kernels, system calls and system programs, Various components of operating system with reference to DOS. BIOS, BIOS and DOS interrupt, Single user operating system. Task loader, Memory management.

Unit 3: Process Management (10 Lectures)

System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-preemptive and preemptive scheduling algorithms; concurrent and processes, critical section, semaphores, methods for inter-process communication; deadlocks. Device management. Control of various devices. Device drivers. Interrupt driven and poll driven data transfers. Need of software and hardware protocols.

Unit 4: Memory Management (08 Lectures)

Physical and virtual address space; memory allocation strategies -fixed and variable partitions, paging, segmentation, virtual memory

Unit 5: File and I/O Management (08 Lectures)

Directory structure, file operations, file allocation methods, device management, File systems. File management. Process management and scheduling Special requirements and facilities for multiprocessing environment

Unit 6: Protection and Security (4 Lectures)

Policy mechanism, Authentication, Internal access Authorization.

Unit 7: Introduction to Unix (08 Lectures)

Brief history of UNIX, Features of UNIX, Structure of the UNIX system, functions of shell & kernel. UNIX File System (etc,bin,dev,usr,lib,tmp).

Types of files (ordinary, special & directory files),

Pathname - absolute & relative, block and character device,

Internal structure of File System (boot block, super block, inode block, data block).

Files opened by UNIX - stdin(0), stdout(1), stderr(2)

Shell variables - HOME, PATH, TERM, LOGNAME, PS1, PS2, MAIL.

Unit 8: Unix Commands

(10 Lectures)

ls [options- A,d,F,i,l,r,R,t,u,x] [files..]wild card characters *, ?, [a..z], cat , rm [options -i,r][files..], cal, date [m,h,a,D,x,V], banner, passwd, pwd, echo, who[options- H,u], who am i, logname, clear, tty, tput, clear, mkdir, cd, rmdir, comm, diff, cmp, chmod, chgrp, chown, Redirection & pipes { <, >, >>, | } cp [options -i,r][files..], tee, ln, mv,wc [options-l,w,c] [files..], grep[options-c,i,l,n,v],pr [(+n)d(on)(1n)] [files....], passwd, pg, more, sleep,write, mesg, wall,lp [cm(n number)], ps [options- a, e, u], kill [-9] <pid>.Vi modes of vi - command mode & edit mode.vi +n file, vi /pattern fileexiting - ZZ, :wq, :q!, :qi, :x , window movement- ^d, ^u, ^f, ^b, z, z., z-,text entry - a, i, o, A, O, I, cursor movement - l , k,j,h,^, \$,[n]w, [n]b, [n]e,), (,return,H, M, Ldeletion of text-[n]x, [n]X,[n]r,R, [n]s,d\$,d^ undo- u & U, redraw screen-^, pattern searching -/pattern, ?patternrearranging textJ, P, p operators- [n]dd, [n]yy, cc, vi options - :set,:set allnu, showmodelast line commands contd ... :w!, :w <name>, :w! <name>, :w, :r <name>

PROGRAMMING THE BOURNE SHELL

echo, \c, \t, \n, \v, \l

executing a shell script

test <expression>

files :r, w,x,f,d,e,s

string :z,n,s l=s2,s l!=s2

numerics :- n1 [opt] n2

opt - gt, lt, eq, ne, ge, le

read <variable>, if ... then..elif..else ... fi,

case ... esac, while, until, for.

Operating Systems Lab[Credit: 2, Lab : 30]

Students are advised to do laboratory/practical practice.

Course Outcome:

After learning the course the students should be able to:

Understand various generations of Operating System and functions of Operating System
Write shell scripts in Linux/UNIX environment.

Reference books :

1. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
2. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education 1997.
3. W. Stallings, Operating Systems, Internals & Design Principles, 5th Edition, Prentice Hall of India. 2008.
4. M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.
5. Operating Systems, A K Sharma, University Press
6. Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.
7. Operating System Concepts (4th Ed.) by Abraham Silberschatz & Peter B. Galvin
8. A User Guide to UNIX System by Dr. Rebecca Thomas and Jean Yates
9. Concepts & Application by Sumitabha Das
10. Peter Norton's guide to Unix

CC-7: Structured System Analysis and Design

[Credit: 6(5+1), Lecture Hours: 60]

Unit 1. Concept of System(10 Lectures)

Characteristics of system (Organisation, Interaction, Interdependence, Integration,Central objectives), Types of system -Physical & abstract system, open & closed system, Man made information system, Computer based information system.

Unit 2.System Life Cycle(10 Lectures)

Different stages of the system life cycle and the forms generated at each level -Conception, initiation, analysis, design, construction (project request form, system proposal, functional specifications, information requirement table) role of the systems analyst.

Unit3.Context Analysis Diagram(10 Lectures)

Introduction to Context analysis diagram, Data flow diagram, External entity, data flow, data store, process

Unit 4.Normalization(05 Lectures)

Normalization first level, second level and third level normalization

Unit 5.Introduction to Data dictionary(05 Lectures)

Data dictionary and its organization,Data structure, data element, data flows, data stores & process (decision tree, decision table, structured english)

Unit 6. DFD(10 Lectures)

Levelled data flow diagrams, Balancing of levelled DFD's, local data stores, current physical DFD, currentlogical DFD, proposed logical DFD,proposed physical DFD, automation boundary

Unit 7.Systems Design(10 Lectures)

Systems Design Translations of functional specifications into design specifications i.e.,from logical design to physical design

Reference Books:

1. Analysis And Design Of Information System. - V. Rajaraman

2nd Year: Semester IV

CC-8: Computer Networks [Credit: 6(5+1), Lecture Hours: 60]

Unit 1: Introduction to Computer Networks (8 Lectures)

Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite.

Unit 2: Data Communication Fundamentals and Techniques (10 Lectures)

Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation-; multiplexing techniques- FDM, TDM; transmission media.

Unit 3: Networks Switching Techniques and Access mechanisms (10 Lectures)

Circuit switching; packet switching- connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.

Unit 4: Data Link Layer Functions and Protocol (10 Lectures)

Error detection and error correction techniques; data-link control- framing and flow control; error recovery protocols-stop and wait ARQ, go-back-n ARQ; Point to Point Protocol on Internet.

Unit 5: Multiple Access Protocol and Networks (5 Lectures)

CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks- repeaters, hubs, switches, bridges, router and gateways;

Unit 6: Networks Layer Functions and Protocols (6 Lectures)

Routing; routing algorithms; network layer protocol of Internet- IP protocol, Internet control protocols

Unit 7: Transport Layer Functions and Protocols (6 Lectures)

Transport services- error and flow control, Connection establishment and release- three way handshaking

Unit 8: Overview of Application layer protocol (5 Lectures)

Overview of DNS protocol; overview of WWW & HTTP protocol

Suggested Readings

1. B. A. Forouzan: Data Communications and Networking, Fourth edition, THM, 2007.
2. A. S. Tanenbaum: Computer Networks, Fourth edition, PHI , 2002
3. Dr. Rakesh Kumar Mandal: Computer Networks for Students, First Edition, SPD, 2018

CC -9 : VB.NET

[Credit: 4, Lecture Hours: 60]

Unit 1 : Introduction – .NET Framework Technology, Advantage, Framework Class Library : data types, exceptions, events, attributes, collections, remote calls, threading, web services etc., Working with .NET FCL : namespaces. **(04 Lectures)**

Unit 2 : Language – Data Types, Operators, Modifiers, Directives, Programming Constructs : decision making, case and iteration, Arrays, Procedures & Functions, Library Functions : string, date & time, math, number and Miscellaneous. **(04 Lectures)**

Unit 3 : OOPS – Features, Declaring classes, Properties, Public & Private variables, Inheritance, Overloading, Overriding, Virtual Methods and Abstract Class. **(05 Lectures)**

Unit 4 : Windows Form – Introduction to Class Libraries, Event and Event Handlers, Winforms and GUI, Creating Applications using Winforms, Concept of Controls, Different GUI Controls, their properties, methods and event procedures and Programs using controls. **(15 Lectures)**

Unit 5 : Error Handling in Windows Form – Introduction to Errors, Exceptions & Validations, Types of Validations, Types of Errors, Types of Exceptions. **(07 Lectures)**

Unit 6 : SDI & MDI Applications – Introduction, Characteristics & Drawbacks, File Handling program. **(03 Lectures)**

Unit 7 : Data Access – Introduction to Microsoft Data Access Technologies, ADO.NET : overview, architecture, class libraries, Creating Database, Connecting to Database, Databound Controls, Creating Data Set, Manipulating Records and XML Data. **(15 Lectures)**

Unit 8 : Component Object Module (COM) – Fundamentals of Custom Control, Introduction to COM, Events and User Controls, Web Programming. **(07 Lectures)**

Core Course 9 - Lab

[Credit: 2, Lab : 30]

Students are advised to do laboratory/practical practice.

Reference Books:

1. Programming in VB .Net: V. Karthika: Books

CC -10 : RDBMS and Oracle [Credit: 4, Lecture Hours: 60]

Unit 1: Introduction

ORACLE, Relational Approach, Various aspect of relational model(04 Lectures)

Unit 2: Introduction to SQL(10 Lectures)

Feature of SQL, Query with SQL, Arithmetic expression, Column aliases, Concatenated columns, Literals Writing SQL statements, Clauses, Select, order by, where, group by, distinct, having, SQL operators, Between... and..., In(list), Like, Is null, Not between, Not in, Not like, Is not null, SQL commands, Del, list, get, save, run, start, edit, exit, clear buffer, etc, Running standard queries with substitution variables, Single ampersand substitution variable, double ampersand substitution variable, define, underline, accept, prompt, no prompt, hide

Unit 3: Functions(10 Lectures)

Date function: ADD_MONTHS, LAST_DAY, MONTHS_BETWEEN, NEXT_DAY, GREATEST, LEAST etc. Formats :MM, DD, MON, MONTH, D, DAY, YYYY, YY, YY, YEAR, WW, W, HH, A.M.,P.M., FM, TH, SP, SPTH, THSP, etc.

String function: RPAD, LPAD, LOWER, UPPER, INITCAP, LENGTH, SUBSTR, INSTR, TRANSLATE, REPLACE

Arithmetic function: ABS, ROUND, COS, LOG, EXP, SQRT, SIGN, SIN, MOD, POWER, FLOOR etc.

Conversion function: TO_CHAR, TO_DATE, TO_NUMBER etc.

Miscellaneous function: DECODE, NVL etc

Group column function: AVG, COUNT, MAX, MIN, SUM etc.

Nested function : Sub queries, nested sub queries

Join: Equi join, outer join, inner join, non-equi join

Set operator: Union, intersect, minus

Unit4: Building a report in SQL*PLUS (10 Lectures)

Commands: Column, rem, heading. Null, skip, ttitle, btitle, etc., Breaking a report in sections, Break on, compute, Set commands, spooling, show commands, environments variables, Creating defining table structures (DDL), Introduction to data definition language, Creating a table, Integrity constraints, Creating tables using hierarchical relation, Creating tables with rows from another table, Alter table & drop table, Data dictionary, Create views, Create sequences, Create index, Data manipulation language(DML), Commands to manipulate data, Data control language (DCL), DCL Commands

Unit 5: PL/SQL (10 Lectures)

Introduction, PL/SQL structure, The environment of PL/SQL, Features of PL/SQL, The rules of block structure (Declare , Begin , ... End), Nested block, Declaring Constants and variables, PL/SQL variables and Bind variables, Procedures, Functions, SQL Commands in PL/SQL, Exception handling, Predefined, User Defined, Control structure, If statement, GOTO, For..Loop, while, exit, Records in

PL/SQL, %ROWTYPE, Implicit & Explicit cursors, cursor, fetch, declare, open, close etc., Database Triggers, Creating Triggers, Before and After Trigger, Row level Trigger, Statement level Trigger

Unit 6: ORACLE FORMS:(10 Lectures)

Introduction, Oracle form designer, Object navigator, Canvas view, Property Sheet, PL/SQL editor, Blocks, LOVs, Radio Buttons, Alerts, Form Triggers, Form Level Triggers, Block level triggers, Item level triggers, Query triggers, Navigation triggers, Writing PL/SQL code, Master Detail relationship

Unit7: ORACLE REPORT DESIGNER(06 Lectures)

Creation of reports, Ordinary reports & matrix reports

Core Course 10- Lab [Credit: 2, Lab: 30]

Development of a business application in RDBMS

3rd Year: Semester V

CC -11 : Data Structures

[Credit: 4, Lecture Hours: 60]

Unit 1: Introduction to data structures

Unit 2 : Linear data structures

Arrays: Definition, concept and operation on arrays

Linked list :Singly linked list, Circularly linked list, Doubly linked list

Stacks: Creation, Push , Pop

Queues: creation, insertion and deletion

Unit 3: Non-linear data structures

Tree: Definition, concept and operation of trees, Storage representation of binary trees, Manipulation of binary trees, B_tree, AVL trees, Tree Traversal Algorithms

Unit4: Sorting

Selection sort, Bubble sort, Merge sort, Quick sort, Heap sort,

Unit 5: Searching

Sequential searching, Binary searching

Reference books:

1. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
2. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
3. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using C and C++:", Second edition, PHI, 2009.
4. Robert L. Kruse, "Data Structures and Program Design in C++", Pearson, 1999.
5. D.S Malik, Data Structure using C++,Second edition, Cengage Learning, 2010
6. Mark Allen Weiss, "Data Structures and Algorithms Analysis in Java", Pearson Education, 3rd edition, 2011.
7. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using Java, 2003.
8. Robert Lafore, "Data Structures and Algorithms in Java, 2/E", Pearson/ Macmillan Computer Pub,2003.
9. John Hubbard, "Data Structures with JAVA", McGraw Hill Education (India) Private Limited; 2 edition, 2009.
10. Goodrich, M. and Tamassia, R. "Data Structures and Algorithms Analysis in Java", 4th Edition, Wiley,2013
11. Herbert Schildt, "Java The Complete Reference (English) 9th Edition Paperback", Tata McGraw Hill, 2014.
12. D. S. Malik, P.S. Nair, "Data Structures Using Java", Course Technology, 2003.

Core Course 11- Lab[Credit: 2, Lab Hours: 60]

Students are advised to do laboratory/practical practice.

CC-12: Introduction to Big Data Analytics

[Credit: 6(5+1), Lecture Hours: 60]

Unit 1 : Overview – Introduction, Data Mining Life Cycle : CRISP-DM Methodology and SEMMA Methodology. Big Data Life Cycle **(04 Lectures)**

Unit 2 : Methodology – Difference from traditional statistical approach of experimental design, Statistical Modelling. **(03 Lectures)**

Unit 3 : Deliverables & Stakeholders in BDA – Machine Learning Implementation, Recommender System, Dashboard, Ad-Hoc Analysis, Project/Project Understanding, Establishing advantages of the analysis. **(15 Lectures)**

Unit 4 : Data Analyst & Data Scientist – Introduction, Work-area and Role. **(01 Lecture)**

Unit 5 : BDA Methods – Introduction to R, Introduction to SQL, Charts and Graphs, Data Tools, Statistical Methods. **(07 Lectures)**

Unit 6 : BDA Project Management – Project Description, Problem Definition : Supervised Classification, Supervised Regression, Unsupervised Learning & Ranking, Data Collection (mini project-twitter), Data Mining / Cleansing : homogenization & heterogenization, Summarizing, Data Exploration : exploratory data analysis and Data Visualization : generation of correlation matrix visualization. **(30 Lectures)**

3rd Year: Semester VI

CC13: Web Technologies [Credit: 4, Lecture Hours: 60]

Unit 1 : Web Basics – Introduction & Concept of Internet, History of Internet, Advantage and Disadvantage of Internet, Protocols Used, WWW, Domain : Letter Addressing System (URL), Domain : Number Addressing System (IP Address), Dedicated Access, Dial-up Access, Purchasing/Renting Domains (web sites), Web Server, Browsers, Introduction to Cyber Law, Introduction to e-Commerce. **(04 Lectures)**

Unit 2 : HTML – Introduction and History, Basic structure of HTML document, Head Elements : Title, Meta, Style, Base & Link. Body Element properties : Background, BGCOLOR, Text, Link, Vlink, Alink, BGProperties & Margin. Font Element properties : Size, Color & Face. Other text basics - Heading, Line Break, Paragraph, Span, HR, Center. Marquee Element properties : Align, Behaviour, BGCOLOR, Direction, Height, Width, Loop, ScrollAmount, ScrollDelay, HSpace, VSpace. Anchor Element properties : Href, Rel, Name, Title & Targets. List Elements and their properties : LI, OL, UL, Menu & DIR. Table elements – Table, TH, TR & TD and their properties – Rowspan, Colspan, Caption, Border, Width, Align, VAlign, BGCOLOR, BorderColor, & CellSpacing. Form Elements – Input, Select & Option and their properties – Type, Value, Name, ID, Size, Checked, MaxLength, Align, SRC, Multiple & Selected. Form properties : Methods – Get & Post and Action. HTML Media. **(15 Lectures)**

Unit 3 : Cascading Style Sheet (CSS3) – Introduction, Units, Types, Basic properties : text, font, color, background, image, link, table, border, margin, list, padding, cursor, scrollbar, dimension, visibility, positioning, layers, text effects, layouts, multi background, shadow, multi columns, gradient, user interface, responsive css, animation etc. **(05 Lectures)**

Unit 4 : Dynamic HTML and XML – Introduction, Difference b/w HTML and DHTML, Advantages of DHTML, CSS for DHTML, Document Object Models (DOMs), Event Handling, Data Binding, Introduction to XML. **(05 Lectures)**

Unit 5 : PHP (with ref to HTML) – Introduction, Program Structure, Data Types, Variables and Constants, Operators, Programming Constructs : Decision, Iteration and Case. Strings, Arrays, Accessing HTML data using Get and Post, User Defined Functions and their usage, Library Functions : String, Date & Time, Number, Math and Miscellaneous. Global Constants and Macros. Sessions, Cookies, File Handling, Data Uploading, Sending Emails and SMSs, Error Handling, OOPS. **(20 Lectures)**

Unit 6 : MySQL (with ref. To PHP) – Introduction, Database operations : creation, selection and dropping and connecting from web-page using PHP. Data Types. Table operations : creating, altering, dropping, inserting/updating/deleting/querying of records, all using PHPMyAdmin and PHP code. **(04 Lectures)**

Unit 7 : Java Script and AJAX (with ref. To PHP and HTML) – JS Introduction, Program Structure, Data Types, Variables and Constants, Operators, Programming Constructs : Decision,

Iteration and Case. Strings and Arrays. User Defined Functions and their usage. Event Handling, Error Handling, Validation and Animation. **AJAX** introduction, XMLHttpRequest, Database operations.
(07 Lectures)

Core Course 13- Lab[Credit: 2, Lab: 30]

Software Lab Based on Web Technologies: Students are advised to do laboratory/practical practice.

CC-14 : Android and App Development

[4 Credits, 60 Lectures]

Unit 1: Refreshing Java:

(01 Lectures)

Javaintroduction, OOPs Concepts, Abstract class & Interface, Exception handling, Multi threading(Thread class & RunnableInterface)

Unit 2: Android:

(08 Lectures)

Introduction to Android, WhyAndroid?, Android RunTime, Android Studio, Introduction toGradle, Fundamentals : Basic Building blocks – Activities, Services, Broadcast Receivers & Contentproviders, UI Components- Views ¬ifications, Components for communication -Intents & IntentFilters, Android API levels(versions & versionnames)

Unit 3: Application Structure:

(09 Lectures)

AndroidManifest.xml, uses-permission, Activity/services/receiverdeclarations, Resources &R.java, Layouts & DrawableResources, Activities and Activitylifecycle

Unit 4: Emulator:

(08 Lectures)

Launching emulator, Editing emulatorsettings, Emulatorshortcuts, Logcatusage, Introduction to Android Device Monitor(ADM), Fileexplorer, Intents: ExplicitIntents, Implicitintents

Unit 5: Basic UI design, Styles & Themes :

(08 Lectures)

Formwidgets, Text Fields, Layouts - RelativeLayout , TableLayout,FrameLayout, LinearLayout, Nested layouts, [dip,dp,sip,sp] versuspx, styles.xml, drawable resources forshapes, gradients(selectors), Style attribute in layoutfile, Applying themes via code and manifestfile, AlertDialogs &Toast, Time and Date, Images andmedia

Unit 6: Menu:

(08 Lectures)

Option menu And Action Bar(menu in actionbar), Context menu and contextual actionmode, Popup menu, menu fromxml, menu via code, Linkify, MatchFilter &TransformFilter, Adapters : ArrayAdapters, BaseAdapters, ListView andListActivity, Customlistview, GridView usingadapters, Gallery usingadapters, Android Session and Sessionmanagement

Unit 7: Content Providers:

(10 Lectures)

SQL, DML & DDL Queries inbrief, SQLiteDatabse, SQLiteOpenHelper, Cursor, SQLiteProgramming, Reading and updatingContacts, Android Debug Bridge(adb)tool, Broadcast Receivers, Services

Unit 8: Notifications:

(08 Lectures)

Alarm, Viaservice, Customize: Toast, Dialogs, Tabs, Animatedpopuppanels, Gridview, Spinner, Thread.AsyncTask, XMLParsing, Android JSON parsing usingVolley, How to create REST API for Android app usingPHP, Mysql, Accessing Phone services(Call,SMS)

Core Course 14- Lab[Credit: 2, Lab: 30]

Software Lab Based on Android: Students are advised to do laboratory/practical practice.