



DETAILED SYLLABUS FOR BACHELOR OF COMPUTER APPLICATION



**DEPT. OF COMPUTER
SCIENCE & ENGINEERING**

SCHEME – SEMESTER WISE COURSE ALLOCATION

First Semester Syllabus							
Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	HSM	BCA101	General English& Soft Skill	3	0	2	4
3	CC	BCA103	Basic Computation & Principles of Computer Programming	3	1	3	5.5
4	DE*		Elective-I	3	0	3	4.5
5	DE**		Elective-II	3	0	3	4.5
6	GE*		Elective-III	3	0	0	3
7	GSC	GSC101	ESP & SDP-I	0	1	0	1
Total							22.5

Students will undergo a mandatory induction program

Suggestive Choice Based Subjects

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	DE*	BCA104A	Introduction to Windows Programming	3	0	3	4.5
2	DE*	BCA104B	Introduction to Python Programming	3	0	3	4.5
3	DE**	BCA105A	Analog Electronics	3	0	3	4.5
4	DE**	BCA105B	Digital Electronics	3	0	3	4.5
5	GE*	BCA106A	E-Commerce	3	0	0	3
6	GE*	BCA106B	Business System & Administration	3	0	0	3

TITLE OF COURSE: BASIC COMPUTATION & PRINCIPLES OF COMPUTER PROGRAMMING

COURSE CODE: BCA103

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Introduction:

Computers are so widely used in our day-to-day lives that imagining a life without them has become almost impossible. Learning computer fundamentals is a stepping stone to having an insight into how these machines work. Once the student is aware of the basic terminology that is commonly used in computer science, he/she can then go on to develop useful computer programs that may help solve a user's problem. Since computers cannot understand human languages, special programming languages are designed for this purpose. C is one such programming language. Being the most popular programming language, it is used in several different software platforms such as system software and application software. A few other programming languages such as C++ and JAVA are also based on C. Hence, mastering the C is prerequisite to become a successful computer engineer.

Objectives:

1. Learn how to solve common types of computing problems.
2. Learn data types and control structures of C
3. Learn to map problems to programming features of C.
4. Learn to write good portable C programs

Course Outcomes (CO):

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

1. Appreciate and understand the working of a digital computer
2. Analyze a given problem and develop an algorithm to solve the problem
3. Improve upon a solution to a problem
4. Use the 'C' language constructs in the right way
5. Design, develop and test programs written in 'C'

Course Contents:

Unit 1: Introduction to Computers – Generations, Classifications, Applications, Basic Organization. Input and output devices. Basic concept of Computer memory, Computer software and networks.

Unit 2: Number system – Decimal, Binary, Octal, Hexa-decimal. Conversion of numbers, Addition and subtraction of two numbers. Two's complement, Multiplication and division of binary numbers. Working with fractions, signed number representation in binary form, Logic gates.

Unit 3: Introduction to C – compiling and executing C programs, using comments, keywords, identifiers, Data type, variables, constants, input/output statements in C, operators in C, type conversion and type casting.

Unit 4: Decision Control and looping statements – conditional branching statement, iterative statements, nested loops, break and continue statements, goto statement.

Unit 5: Arrays – Declaration, accessing elements of array, storing values, calculating the length of array, two dimensional arrays. Strings – reading and writing strings, suppressing input, string taxonomy, string operations – using and without using library function, array of strings.

Unit 6: Functions – Declaration, prototype, definition, function call, return statement, passing parameters to the function, scope of variable, storage classes, recursive functions.

Unit 7: Pointers – introduction, declaration, Pointer expression and arithmetic, null pointer, generic pointer, passing arguments to functions using pointer, pointers and arrays, passing an array to function, difference between array name and pointer, pointers and strings, array of pointers, function pointers, pointers to pointers, dynamic memory allocation, drawbacks of pointers.

Unit 8: Structure, nested structure, array of structure, union, array of union variable, unions inside structure. Files – Reading –writing etc. Preprocessor directives.

Text Books

1. Brian Kernighan and Dennis Ritchie, The C Programming Language, 2nd Edition, Prentice Hall PTR, 1988.
2. ReemaThareja, Computer fundamentals and Programming in C, Oxford university press, 2012

TITLE OF COURSE: BASIC COMPUTATION & PRINCIPLES OF COMPUTER PROGRAMMING LAB

COURSE CODE: BCA193

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Introduction:

This course is designed to familiarize students with the basic components of a computer, so as to be able to operate it and be able to interact with it, and carry out simple tasks. In addition, it

will initiate the students into the discipline of Programming. It aims to start off the development of problem solving ability using computer programming. This course teaches not only the mechanics of programming, but also how to create programs that are easy to read, maintain, and debug. Students are introduced to the design principles for writing good programs regardless of the hardware and the software platforms.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of software development and maintenance. To reach this goal, the following objectives need to be met:

1. Understand the basic terminology used in computer programming
2. Write, compile and debug programs in C language.
3. Use different data types in a computer program.
4. Design programs involving decision structures, loops and functions.
5. Explain the difference between call by value and call by reference
6. Understand the dynamics of memory by the use of pointers.
7. Enhance programming skills through problem solving and code development of small-size software applications.
8. Improve self-learning, teamwork and communication skills through project development practices.
9. Engage in continuing professional development under minimal guidance.

List of Experiments

1. Write a program to evaluate area of triangle using the formula $\sqrt{s(s-a)(s-b)(s-c)}$
2. Write a program to swap two numbers.
3. Write a program to find the greatest of three numbers and print the numbers in ascending order.
4. Write a program to perform the arithmetic expression using switch statement.
5. Write a program to find a factorial of given number using do while statement.
6. Write a program to print all prime numbers upto 'N' numbers.
7. Write a program to print sum of 'N' natural numbers.
8. Write a program to find the total number of even integers and odd integers of 'N' numbers.
9. Write a program to find the sum of odd numbers and even numbers upto 'N' numbers.
10. Write a program to print the product of two matrices of any order.
11. Write a program to read 'N' number of students with 5 subject marks.
12. Write a program to find greatest of 'n' numbers using functions.
13. Write a program to print Fibonacci series using recursion.
14. Write a program to convert all lower case to uppercase characters.
15. Write a program to sort 5 city names in alphabetical order.
16. Write a program to extract a string.
17. Write a program to implement the concept of call by value.
18. Write a program to implement the concept of call by reference.
19. Write a program to implement the concept of structure and union.
20. Write a program to access a variable using pointer.
21. Write a program to print the element of array using pointers.
22. Write a program to print the elements of a structure using pointers.

23. Write a program to display student information by initializing structures.
24. Write a program to pass structure as arguments to function and calculate total marks of 5 subjects.
25. Write a program to write integer data into file and read it from file.

References

1. Daniel Appleman, How Computer Programming Works, Apress, 2000.
2. Harvey M. Deitel & Paul J. Deitel, C How to Program, Fourth Edition.
3. Delores M. Etter, Engineering Problem Solving with C, Third Edition, Prentice Hall, 2005.
4. Brian W. Kernighan & Rob Pike, the Practice of Programming, Addison-Wesley, 1999.
5. Stefano Ceri, Dino Mandrioli & Licia Sbattella, The Art and Craft of Computing, Addison-Wesley, 1998,

TITLE OF COURSE: GENERAL ENGLISH & SOFT SKILL

COURSE CODE: BCA101

L-T-P: 3-0-2

CREDITS: 4

SYLLABUS

Introduction:

Hone the English speaking and writing skills with this Basic English structure and soft skills.

Course Outcomes (CO):

It is a well-balanced course that focuses on the four core language skills: listening, speaking, reading and writing. The objective of our General English Course is to help you to use the English language more effectively.

Course Content:

Unit I: Vocabulary Building

1. The concept of Word Formation
2. Root words from foreign languages and their use in English
3. Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives.
4. Synonyms, antonyms, and standard abbreviations.

Unit II: Basic Writing Skills

1. Sentence Structures
2. Use of phrases and clauses in sentences
3. Importance of proper punctuation
4. Creating coherence
5. Organizing principles of paragraphs in documents
6. Techniques for writing precisely

Unit III: Identifying Common Errors in Writing

1. Subject-verb agreement
2. Noun-pronoun agreement
3. Misplaced modifiers
4. Articles
5. Prepositions
6. Redundancies
7. Cliches

Unit IV: Nature and Style of sensible Writing

1. Describing
2. Defining
3. Classifying
4. Providing examples or evidence

Unit V: Writing introduction and conclusion

1. Writing Practices
2. Comprehension
3. Precis Writing
4. Essay Writing

Unit VI: Oral Communication (This unit involves interactive practice sessions in Language Lab)

1. Listening Comprehension
2. Pronunciation, Intonation, Stress and Rhythm
3. Common Everyday Situations: Conversations and Dialogues
4. Communication at Workplace
5. Interviews
6. Formal Presentations

TEXT BOOKS:

1. On writing: a memoir of the craft by stephen king
2. Essential english grammar; by helennaylor (author), raymond murphy (author)

TITLE OF COURSE: INTRODUCTION TO WINDOWS PROGRAMMING**COURSE CODE: BCA104A****L-T-P: 3-0-3****CREDITS: 4.5**

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in C Programming, Other programming knowledge.

Introduction:

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

Course Outcomes (CO):

Students will learn to evaluate engineering problems, formulate one or more solution techniques or algorithms, and code the solution using Microsoft Visual Basic for Applications (VBA) software and connect database with Microsoft Access.

1. Student learns about Visual Basic's Integrated Development Environment (IDE).
2. Student write Visual Basic programs using object-oriented programming techniques including
3. Classes, objects, methods, instance variables, composition, and inheritance, and polymorphism.
4. Understand windows applications using forms, controls, and events.
5. Understand design, create, build, and debug Visual Basic applications.
6. Understand database design, connectivity with backend tool.
7. Create one and two dimensional arrays for sorting, calculating, and displaying of data.
8. Write and apply procedures, sub-procedures, and functions to create manageable code.
9. Write and apply decision structures for determining different operations.
10. Analyze a given problem and implement an algorithm to solve the problem.
11. Improve upon a solution to a problem.
12. Implement the Visual Basic language constructs in the right way.

Course Contents:

Module-1: Introduction – Installing Visual Basic, First Application, Toolbox, Property Editor, VB Forms, buttons, Events, Adding Code, Dealing with Errors, Project Explorer, Saving and Reopening Applications.

Module-2: Controls Explained – Controls and Methods, Using Buttons, Labels, Text Boxes, Picture Boxes, Check Boxes, Option Buttons, Frames, Lines and Shapes, Images, List Boxes, Flex Grids, CommonDialog Control, Combo Boxes, Timers, OLE Control, Setting Tab Order.

Module-3: Basics – Variables and Scope, Data Types, Using If...Then...Else, For...Next, Do...Loop, Case...Else, With...End With, Input Box, API, Printing. VB Tools – Menus, Toolbars, Code Editor, ActiveX Controls, Reference Manager, Project Templates.

Module-4: Introduction to Databases, Visual Data Manager, Designing Table, Adding Data, Adding Records, Deleting Records, Viewing Records, Searching Records, Showing Records in Grid, Copying Record to Clipboard.

Module-5: VB and Internet – Basics of Web Scripting.

Text Books

1. Greg Perry, SnajayaHettihewa, “SAMS Teach Yourself Visual Basic 6 in 24 Hours”, Pearson Education.
2. Tim Anderson, “Visual Basic 6 in Easy Steps”, Dreamtech Press.

References

1. Steven Holzner, “Visual Basic 6 Programming Black Book”, Dreamtech Press.

TITLE OF COURSE: INTRODUCTION TO WINDOWS PROGRAMMING LAB**COURSE CODE: BCA194A****L-T-P: 0-0-3****CREDITS: 1.5**

SYLLABUS

Prerequisite: Students must have already registered for the course, “Software Engineering”.

Objectives:

Students will learn to evaluate engineering problems, formulate one or more solution techniques or algorithms, and code the solution using Microsoft Visual Basic for Applications (VBA) software and connect database with Microsoft Access. Professionalism in completing and presenting laboratory exercises is emphasized.

Course Outcomes (CO):

Student learns about Visual Basic’s Integrated Development Environment (IDE). Student write Visual Basic programs using object-oriented programming techniques including classes, objects, methods, instance variables, composition, and inheritance, and polymorphism.

1. Understand windows applications using forms, controls, and events.
2. Understand design, create, build, and debug Visual Basic applications.
3. Understand database design, connectivity with backend tool.
4. Create one and two dimensional arrays for sorting, calculating, and displaying of data.
5. Write and apply procedures, sub-procedures, and functions to create manageable code.
6. Write and apply decision structures for determining different operations.
7. Analyze a given problem and implement an algorithm to solve the problem.
8. Improve upon a solution to a problem.
9. Implement the Visual Basic language constructs in the right way.
10. Design, develop and test Applications written in Visual Basic.

Course Contents:

Exercises that must be done in this course are listed below:

1. Controls & Properties. Idea about Labels, Buttons, Text Boxes.
2. Data basics, Different type variables & their use in VB,
3. Sub-functions & Procedure details, Input box () &Msgbox ().
4. Making decisions, looping
5. List boxes & Data lists, List Box control, Combo Boxes, data Arrays.
6. Frames, buttons, check boxes, timer control,
7. Programming with data, ODBC data base connectivity.
8. Data forms Wizard, query, and menus in VB Applications,
9. Graphics.

Text Book:

1. Greg Perry, SnajayaHettihewa, “SAMS Teach Yourself Visual Basic 6 in 24 Hours”, Pearson Education.
2. Tim Anderson, “Visual Basic 6 in Easy Steps”, Dreamtech Press.

Recommended Systems/Software Requirements:

1. Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
2. Turbo C or TC3 compiler in Windows XP or Linux Operating System.

TITLE OF COURSE: INTRODUCTION TO PYTHON PROGRAMMING**COURSE CODE: BCA104B****L-T-P: 3-0-3****CREDITS: 4.5**

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and basic mathematics.

Introduction:

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

Course Outcomes (CO):

After completion of the course, students will able:

1. Develop algorithmic solutions to simple computational problems
2. Read, write, execute by hand simple Python programs.
3. Structure simple Python programs for solving problems.
4. Decompose a Python program into functions.
5. Represent compound data using Python lists, tuples, and dictionaries.
6. Read and write data from/to files in Python Programs.

Course Contents:

Module-1: ALGORITHMIC PROBLEM SOLVING

Algorithms, building blocks of algorithms (statements, state, control flow, functions), notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, and guess an integer number in a range, Towers of Hanoi.

Module-2: DATA, EXPRESSIONS, STATEMENTS

Python interpreter and interactive mode; values and types: int, float, boolean, string, and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.

Module-3: CONTROL FLOW, FUNCTIONS

Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, break, continue, pass; Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.

Module-4: LISTS, TUPLES, DICTIONARIES

Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing - list comprehension; Illustrative programs: selection sort, insertion sort, mergesort, histogram.

Module-5: FILES, MODULES, PACKAGES

Files and exception: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file.

Text Books

1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist“, 2nd edition, Updated for Python 3, Shroff/O‘Reilly Publishers.
2. Guido van Rossum and Fred L. Drake Jr, —An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

References

1. John V Guttag, —Introduction to Computation and Programming Using Python“, Revised and expanded Edition, MIT Press , 2013
2. Robert Sedgewick, Kevin Wayne, Robert Dondero, —Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
3. Timothy A. Budd, —Exploring Python, Mc-Graw Hill Education (India) Private Ltd., 2015.
4. Kenneth A. Lambert, —Fundamentals of Python: First Programs, CENGAGE Learning, 2012.
5. Charles Dierbach, —Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
6. Paul Gries, Jennifer Campbell and Jason Montojo, —Practical Programming: An Introduction to Computer Science using Python 3, Second edition, Pragmatic Programmers, LLC, 2013.

TITLE OF COURSE: INTRODUCTION TO PYTHON PROGRAMMING LAB

COURSE CODE: BCA194B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and basic mathematics.

Introduction:

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

Course Outcomes (CO):

After completion of the course, students will able:

1. Write, test, and debug simple Python programs.
2. Implement Python programs with conditionals and loops.
3. Develop Python programs step-wise by defining functions and calling them.
4. Use Python lists, tuples, dictionaries for representing compound data.

5. Read and write data from/to files in Python.

Course Contents:

1. Compute the GCD of two numbers.
2. Find the square root of a number (Newton's method)
3. Exponentiation (power of a number)
4. Find the maximum of a list of numbers
5. Linear search and Binary search
6. Selection sort, Insertion sort
7. Merge sort
8. First n prime numbers
9. Multiply matrices
10. Programs that take command line arguments (word count)
11. Find the most frequent words in a text read from a file
12. Simulate elliptical orbits in Pygame
13. Simulate bouncing ball using Pygame

PLATFORM NEEDED

Python 3 interpreter for Windows/Linux

Text Books

1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist``, 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers.
2. Guido van Rossum and Fred L. Drake Jr, —An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

References

1. John V Guttag, —Introduction to Computation and Programming Using Python“, Revised and expanded Edition, MIT Press , 2013
2. Robert Sedgewick, Kevin Wayne, Robert Dondero, —Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
3. Timothy A. Budd, —Exploring Python®, Mc-Graw Hill Education (India) Private Ltd.,, 2015.
4. Kenneth A. Lambert, —Fundamentals of Python: First Programs®, CENGAGE Learning, 2012.
5. Charles Dierbach, —Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
6. Paul Gries, Jennifer Campbell and Jason Montojo, —Practical Programming: An Introduction to Computer Science using Python 3®, Second edition, Pragmatic Programmers, LLC, 2013.

TITLE OF COURSE: ANALOG ELECTRONICS

COURSE CODE: BCA105A

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Basic knowledge about components R, L, C, Network Theorems (Kirchoffs law, Thevenin's theorem, Miller theorem etc.). Basic knowledge about the operation of semiconductor devices (Transistor, Diode, UJT, SCR etc.), Ohms Law, Voltage current equations. Basic knowledge of Differentiation, Integration, Differential equation, matrix etc.

Introduction:

Current Voltage equation. Direction of currentflow. Device limitations, Power consumptions and their limits, usage of appropriate device in the problem. Device selection and comparison, advantages and disadvantages.

Course Outcomes (CO):

Students will be able to design simple circuits with transistor, op-amp, amplifiers, oscillators etc. They will be able to test, repair, modify and take-up design exercise. They will have clear knowledge of basic circuit analysis and its functions and their limitations. Most importantly they will be able to recognize, understand, modify and repair majority of circuits used in professional equipment design.

1. Analyze and appreciate the working of electronic circuits involving applications of diodes and transistors.
2. Comprehend working of amplifiers.
3. Design simple analog circuits using general purpose op-amp IC741.
4. Design combinational digital circuits to meet a given specification using digital ICs
5. Develop simple projects based on the different devices studied in this course.

Course Contents:

Module-1: Filters and Regulators: Capacitor filter, π -section filter, ripple factor, series and shunt voltage regulator, percentage regulation, 78xx and 79xx series, concept of SMPS.

Transistor Biasing and Stability: Q-point, Self-Bias-CE, Compensation techniques, h-model of transistors. Expression for voltage gain, current gain, input and output impedance, trans-resistance & trans-conductance; Emitter follower circuits, High frequency model of transistors.

Module-2: Transistor Amplifiers: RC coupled amplifier, functions of all components, equivalent circuit, derivation of voltage gain, current gain, input impedance and output impedance, frequency response characteristics, lower and upper half frequencies, bandwidth, and concept of wide band amplifier. Feedback Amplifiers & Oscillators: Feedback concept, negative & positive feedback, voltage/current, series/shunt feedback, Barkhausen criterion, Colpitts, Hartley's, Phase shift, Wein bridge and crystal oscillators.

Module-3: Operational Amplifier: Ideal OPAMP, Differential Amplifier, Constant current source (current mirror etc.), level shifter, CMRR, Open & Closed loop circuits, importance of feedback loop (positive & negative), inverting & non-inverting amplifiers, voltage follower/buffer circuit. Applications of Operational Amplifiers: adder, integrator & differentiator, comparator, Schmitt Trigger. Instrumentation Amplifier, Log & Anti-log amplifiers, Transconductance multiplier, Precision Rectifier, voltage to current and current to voltage converter, free running oscillator.

Module-4: Power amplifiers – Class A, B, AB, C, Conversion efficiency, Tuned amplifier

Multivibrator – Monostable, Bistable, Astable multivibrators; Monostable and astable operation using 555 timer. Special Functional Circuits: VCO and PLL.

Text Books

1. Microelectronic Circuits–Sedra and Smith(Fifth Edition) (Oxford)
2. Electronic Devices and Circuit theory–Boylestad and Nashesky–PHI/Pearson Education
3. Millman and Halkias–Integrated Electronics–TMH Op Amp and Linear Ics.
4. Electronics–fundamental—D Chattopadhyaya & P. C. Rakhit
5. Linear integrated circuits-D.Roy Choudhury, Shail B. Jain

References

1. R. A. Gackward–PHI/Pearson Education
2. Sergio Franco–Operational Amplifier (JMH)
3. Electronics Devices and Circuits----S Salivahanan N. Sureshkumar A. Vallavaraj

TITLE OF COURSE: ANALOG ELECTRONICS LAB

COURSE CODE: BCA195A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students must have Knowledge basic concepts of electronics.

Objectives: Students will be capable to acquire Digital Electronics circuit. The use of Digital Kit, Integrated Circuit, LED, implement complete Circuit. They also able to ensure Data and Number System, Boolean algebra Combinational Circuit, Sequential Circuit.

Course Outcomes (CO):

1. Students will also be able to design simple circuits with transistor, op-amp, amplifiers, oscillators etc. They will be able to test, repair, modify and take-up design exercise.
2. They will have clear knowledge of basic circuit analysis and its functions and their limitations.
3. Most importantly they will be able to recognize, understand, modify and repair majority of circuits used in professional equipment design.

Course Contents:

Module-1: Implementation of Diode as clipper & clamper, Zener diode as a voltage regulator.

Module-2: Implementation of ripple and regulation characteristics of full wave rectifier without and with capacitor filter, curves of B.J.T & F.E.T

Module-3: class A, class B & class C power amplifiers, Push-Pull amplifiers.

Module-4: timer circuit using NE555 & configuration for monostable & astable multi-vibrator.

Module-5: Ring counter and Johnson's counter

List of Experiments

1. Study of Diode as clipper & clamper
2. Study of Zener diode as a voltage regulator
3. Study of ripple and regulation characteristics of full wave rectifier without and with capacitor filter
4. Study of characteristics curves of B.J.T & F.E.T .
5. Design a two-stage R-C coupled amplifier & study of its gain & Bandwidth.
6. Study of class A & class B power amplifiers.
7. Study of class C & Push-Pull amplifiers.
8. Realization of current mirror & level shifter circuit using Operational Amplifiers.
9. Study of timer circuit using NE555 & configuration for monostable & astable multi-vibrator.
10. Design a Bistable multi-vibrator using NE 555.
11. Study of Switched Mode Power Supply & construction of a linear voltage regulator using regulator IC chip.
12. Design a simple function generator using IC.
13. Realization of a V-to-I & I-to-V converter using Op-Amps.
14. Realization of a Phase Locked Loop using Voltage Controlled Oscillator (VCO).
15. Study of D.A.C & A.D.C.

References

1. Microelectronic Circuits–Sedra and Smith(Fifth Edition) (Oxford)
2. Electronic Devices and Circuit theory–Boylestead and Nashesky–PHI/Pearson Education
3. Millman and Halkias–Integrated Electronics–TMH Op Ampand Linear Ics.
4. Electronics–fundamental—D Chattopadhaya& P. C. Rakhit
5. Linear integrated circuits-D.RoyChoudhury,ShailB.Jain
6. R. A. Gackward–PHI/Pearson Education
7. Sergio Franco–Operational Amplifier (JMH)
8. ElectronicsDevices and Circuits----S SalivahananN.SureshkumarA.Vallavaraj

TITLE OF COURSE: DIGITAL ELECTRONICS

COURSE CODE: BCA105B

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Introduction:

Digital electronics or digital (electronic) circuits are electronics that operate on digital signals. In contrast, analog circuits manipulate analog signals whose performance is more subject to manufacturing tolerance, signal attenuation and noise. Digital techniques are helpful because it is a lot easier to get an electronic device to switch into one of a number of known states than to accurately reproduce a continuous range of values.

Objectives:

1. To acquire knowledge on basics of digital circuits and its applications.
2. This course deals with the basics of Boolean algebra, Digital principles and circuits.
3. The course starts with the basics of Boolean algebra and Boolean expression minimization techniques. Then it explains simple combinational networks like Multiplexers, decoders etc.
4. Sequential and combinational digital circuits are the building blocks of any processor, irrespective of its application.
5. After this the difference between the combinational technologies and sequential circuits is dealt with. Finally, it gives the method to realize the basic gates using different technologies.

Course Outcomes (CO):

Once the student has successfully completed this course, he/she will be able to answer the following questions or perform following activities:

1. Able to explain the basic concepts of digital electronics circuits
2. Able to describe different types of logics, complexity, circuit specifications.
3. On successful completion of this Course, the students would be able to minimize functions using any type of minimizing algorithms (Boolean algebra, Karnaugh map).
4. Define the problem (Inputs and Outputs), write its functions. Implement functions using digital circuit (Combinational or Sequential) and knowledge in analyzing and designing procedures of Combinational and Sequential circuits.
5. To be able to differentiate electronic from electrical systems and identify the basic blocks in any electronic system

Course Contents:

1. Data and number systems; Binary, Octal and Hexadecimal representation and their conversions; BCD, ASCII, EBCDIC, Gray codes and their conversions; Signed binary number representation with 1's and 2's complement methods, Binary arithmetic [5]
2. Venn diagram, Boolean algebra; Various Logic gates- their truth tables and circuits; Representation in SOP and POS forms; Minimization of logic expressions by algebraic method, K-map method [6]
3. Combinational circuits- Adder and Subtractor circuits; Applications and circuits of Encoder, Decoder, Comparator, Multiplexer, De-Multiplexer and Parity Generator. [8]
4. Memory Systems: RAM, ROM, EPROM, EEROM; Design of combinational circuit using ROM, Programming logic devices and gate arrays. (PLAs and PLDs) [5]
5. Sequential Circuits- Basic memory element-S-R, J-K, D and T Flip Flops, various types of Registers and counters and their design, Irregular counter, State table and state transition diagram, sequential circuits design methodology.[9]
6. Different types of A/D and D/A conversion techniques; Logic families- TTL, ECL, MOS and CMOS, their operation and specifications. [6]

Text Books

1. A.Anand Kumar, Fundamentals of Digital Circuits- PHI
2. A.K.Maini- Digital Electronics- Wiley-India
3. Kharate- Digital Electronics- Oxford

TITLE OF COURSE: DIGITAL ELECTRONICS LAB

COURSE CODE: BCA195B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Knowledge is also assumed of basic concepts of electronics.

Objectives:

Students will be capable to acquire about Digital Electronics circuit. The use of Digital Kit, Integrated Circuit, LED, implement complete Circuit. They also able to ensure Data and Number System, Boolean algebra Combinational Circuit, Sequential Circuit,

Course Outcomes (CO):

Students will also be able to handle Digital component using IC. After completion of course student will be able to implement Digital Electronics component.

Course Contents:

Module-1: Implementation of Basic gates and Universal logic gates.

Module-2: Implementation of Half adder, Full adder, Halfsubtractor, Full subtractor, Parallel adder, Parallel subtractor, 1's complement, 2's complement, n'th complement, (n-1)'s complement

Module-3: BCD, 7 segment, Gray code, Binary, Octal, Hexadecimal number system,

Module-4: Flip Flop (SR, JK, D,T)

Module-5: Ring counter and Johnson's counter

List of Experiments

1. Realization of basic gates using Universal logic gates (IC 7408, IC7432, IC7486, IC7404, IC7400, 7402).
2. Implementation of Half adder, Full adder, Halfsubtractor, Full subtractor using basic gates.
3. Code conversion circuits- BCD to Gray code and vice-versa using basic gates.
4. Four-bit parity generator and comparator circuits using basic gates.
5. Construction of simple Decoder and Multiplexer circuits using logic gates.
6. Design of combinational circuit for BCD to decimal conversion to drive 7-segment display using multiplexer.
7. Construction of parallel -Adder, Subtractor using basic gates.
8. Realization of RS-JK and D flip-flops using Universal logic gates.
9. Realization of Universal Register using JK flip-flops and logic gates.
10. Realization of Universal Register using multiplexer and flip-flops.
11. Construction of Adder circuit using Shift Register and full Adder.
12. Realization of Asynchronous Up/Down counter.
13. Realization of Synchronous Up/Down counter.
14. Design of Sequential Counter with irregular sequences.
15. Realization of Ring counter and Johnson's counter using basic gates.
16. Construction of adder circuit using Shift Register and full Adder.

References

1. S. Salivahanan, S. Arivazhagan "Digital Circuit and Design", Vikaspulishin house PVT LTD
2. A.Anand Kumar, Fundamentals of Digital Circuits-PHI

3. A.K.Maini-Digital Electronics-Wiley-India
4. Kharate-Digital Electronics-Oxford

TITLE OF COURSE: E-COMMERCE

COURSE CODE: BCA106A

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is required in Basic internet and the importance of it. Students must be aware of online transactions.

Introduction:

Electronic commerce or ecommerce is a term for any type of business, or commercial transaction that involves the transfer of information across the Internet. It covers a range of different types of businesses, from consumer based retail sites, through auction or music sites, to business exchanges trading goods and services between corporations. It is currently one of the most important aspects of the Internet to emerge.

Course Outcomes (CO):

In this course we will study the basic components of an E-Commerce, their functions, mechanisms, policies and techniques used in their implementation and examples from popular E-Commerce Application. The way different modules in the E-Commerce interact and work together to provide the basic services of an E-Commerce.

1. Understand the theory and logic behind the design and construction of E-Business.
2. You will differentiate between various E-commerce functionalities in terms of performance.
3. Become aware of the issues in the management of resources like EDI, SET, RSA etc.
4. Know the problems in the design of E-Commerce and study the probable solutions.
5. Understanding various type of Business policies.
6. An overview of advanced E-Commerce and compare the technical aspects of all the advanced E-Commerce.
7. To develop, implement, and debug various VB & ASP.
8. To develop, implement, and demonstrate the algorithms of EDI.
9. To develop E-Business.

Course Contents:

Module-1: Definition, Scope of E-Commerce, Hardware requirements, E-Commerce and Trade Cycle, Electronic Markets, Electronic Data Interchange and Internet Commerce.

Module-2: Electronic Markets, Electronic Data Interchange (EDI): Technology, Standards (UN/EDIFACT), Communications, Implementations, Agreements, Security, EDI and Business, Inter-Organizational Ecommerce.

Module-3: Risks: Paper Document vs. Electronic document, Authentication of Electronic document, Laws, Legal issues for Internet Commerce: Trademarks and Domain names, Copyright, Jurisdiction issues, Service provider liability, Enforceable online contract.

Module-4: Security Solutions: Symmetric and Asymmetric Cryptosystems, RSA, DES, and Digital Signature, Protocols for secure messaging, Secure Electronic Transaction (SET) Protocol, Electronic cash over internet, Internet Security.

Module-5: Consumer trade transaction, Internet, Page on the Web, Elements of E-Commerce with VB, ASP, SQL.

Module-6: Internet book shops, Software supplies and support, Electronic Newspapers, Internet Banking, Virtual Auctions, Online Share Dealing, Gambling on the net, E-Diversity, Case studies through internet.

Text Books:

1. E- Commerce-Strategy, Technologies & Applications by David Whitley, TMH.
2. Beginning E-Commerce with VB, ASP, SQL Server7.0 & MTS by Mathew Reynolds, Wrox Publishers.

References:

1. E-Commerce-The cutting edge of business by KamleshK.Bajaj, TMH.

TITLE OF COURSE: BUSINESS SYSTEM & ADMINISTRATION

COURSE CODE: BCA106B

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Introduction:

Business system application, especially e-business systems, use computer and web-based technology to deliver existing business models or promote new ones. This module examines existing business systems, applications and environments, as well as emerging ones that support these business models and system management. The primary focus is on the various business systems, applications and management and understanding how an enterprise framework and integration of disparate application environments facilitate functioning of business and enable development of new models.

Objectives:

The objective of this course is to develop in students an understanding of the concepts, skills and techniques required to become an effective systems analyst who will work with others to create information systems for businesses

Course Outcomes (CO):

On successful completion of this module, students should be able to:

1. Fully understand how e-business systems can contribute to broader enterprise management issues
2. Comprehensively evaluate and critically analyze various business systems (including e-business systems), applications and management
3. Compare and evaluate alternative business application environments that enable business systems
4. Demonstrate a good understanding of architectural options for implementing and facilitating business systems and management
5. Research and evaluate interdependencies between business systems and the rest of the enterprise management environment
6. Critically analyse the impact of business systems and applications on enterprise performance and development of society.
7. Demonstrate interpersonal skills and the ability to work effectively with others to achieve common goals.

Course Contents:

UNIT-1: Use of computers for managerial applications, Technology issues and data processing in organizations, Introduction to Information Systems, shift in Information system thinking, latest trends in Information Technology.

UNIT-2: Computer Based Information Systems- office automation systems. Decision making and MIS, transaction processing systems.

UNIT-3: Decision support system, Group Decision Support, Executive Information systems, DSS generator.

UNIT-4: Introduction to: Artificial Intelligence Based Systems, End user computing, distributed data processing.

UNIT-5: Deciding on IS architecture, IT leadership & IS strategic planning.

UNIT-6: Introduction to: IS strategy and effects of IT on competition.

UNIT-7: Introduction to: ERP, re-engineering work processes for IT applications, Business Process Redesign.

UNIT-8: Knowledge engineering and data warehouse.

Text Books:

1. Management Information System, O'Brien, TMH
2. Management Information System: A Concise Study, Kelkar, PHI
3. Decision support Systems, Janaki Raman, PHI
4. Business Information Systems, Munish Kumar, VIKAS
5. Business Application of Computers, M.M. Oka, EPH

TITLE OF COURSE: ESP & SDP-I**COURSE CODE: GSC101****L-T-P: 0-1-0****CREDITS: 1****Introduction:**

The course is on Quantitative Aptitude, Logical Mental ability, Objective English and Data Interpretation level-I.

Course Contents:**Module-1****Quantitative Aptitude**

- 1) Quant foundation- Vedic Maths & Collective tricks.
- 2) Basic Multiplication – multiplying by numbers ending in zeroes, Multiplying by 2,3,4,5,6,7,8,9, 11,12 & 111. Multiplying 2 digits numbers ending in 9 & whose tens digit at to 10, Multiplying by 2 digits number of 9, multiplying by any 2 digit numbers ending in 9
- 3) Division- Divisibility by 2,3,4,5,6,7,8, 9, 11 & 13; Dividing by 5,9, 15,25,125, Dividing by Factors.
- 4) Squaring numbers- squaring any 2 digit numbers ending in 5, squaring any number ending in 5, squaring any 3 digit numbers ending with 25, squaring any numbers ending in 9, squaring any numbers consisting only nines. Squaring any 2 digit numbers. Cube & cube roots.
- 5) Percentage- Basic concept of percentage & its shortcut rules & their applications.
- 6) Ratio- Basic concept of Ratio & Proportion, Shortcut tricks & their applications.
- 7) Simple equation- Linear equation of 2 & more than two variables.
- 8) Variation- Ratio, Proportion, Variation, concept of directly proportional &
- 9) Partnership – concept, rules & Applications, Percentage Advanced problems & shortcuts.
- 10) Profit & Loss- Basic concept, formulae, shortcut tricks & their Application.

Module-2**Logical Mental ability -1**

- 1) Coding and Decoding & Direction Sense
 - a) Conditional Coding, b) Word-Pattern Coding, c) Chinese Coding, d) Direction Sense Test, e) Direction Distance Test, f) Shadow based Questions
- 2) Series & Numbers
 - a) Alphabet Series, b) Random Series, c) Number Series, d) Letter Gap, e) Missing Number Series, f) Series

Completion, g) Order and Ranking, h) Interchange, i) Comparison

3) Blood Relations

Family Tree Questions, Indication Type BR, Coding Blood Relations, Miscellaneous Blood Relations

4) Analogy

Word Analogy, Classification, Odd-Out

SKILL DEVELOPMENT FOR PROFESSIONALS I

Module-1

Objective English-1

1) Introduction of Parts of speech: Introduction, Brief discussion of Parts of speech

2) What is noun, Kinds of Noun, Rules & Application?

3) Definition of Pronoun, Examples, Rules & Application

4) Definition of Verb, Kinds of Verb, Rules & Application, Definition of Tense, Different types of Tenses, Examples, Rules & Application

5) Definition of Adjective, Kinds of Adjective, Rules & Application,

6) Definition of Adverb, Kinds of Adverb, Rules & Application

7) Definition of Preposition, Examples, Rules & Application,

8) Definition of Interjection, Examples, Rules & Its Application,

9) Definition of Conjunction, Examples, Rules & Application

10) Different types of Articles, Examples, Rules & Application English Grammar.

Newspaper reading: The Hindu & Economic Times.

Module-2

Data Interpretation level-I.

Second Semester Syllabus

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	HSM	BCA201	Professional Writing& Technical Seminar	3	0	3	4.5
2	BSC	BCA202	Mathematics - I	3	0	0	3
3	CC	BCA203	Object Oriented Programming using C++	3	0	3	4.5
4	DE*		Elective-I	3	0	3	4.5
5	DE**		Elective-II	3	0	3	4.5
6	GE*		Generic Elective	3	0	0	3
7	PTI	INT201	Inter/Intra Institutional Activity	0	0	4	2
8	GSC	GSC201	ESP & SDP-II	0	1	0	1
Total							27

Students have to take one mandatory NPTEL course-based on the respective year's offered courses.

Suggestive Choice Based Subjects

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	DE*	BCA204A	Introduction to Linux	3	0	3	4.5
2	DE*	BCA204B	Computer Fundamentals & Emerging Technology	3	0	3	4.5
3	DE**	BCA205A	Simulation & Modeling	3	1	3	5.5
4	DE**	BCA205B	IT & Software Testing	3	1	3	5.5
5	GE*	BCA206A	Environment & Ecology	3	0	0	3
6	GE*	BCA206B	Management Accounting	3	0	0	3

TITLE OF COURSE: PROFESSIONAL WRITING & TECHNICAL SEMINAR**COURSE CODE: BCA201****L-T-P: 3-0-3****CREDITS: 4.5**

SYLLABUS**Introduction:**

This course will help the students to get idea about the professional world.

Objectives:

The course is designed to provide students and faculty with a means for assessment of technical competence of students as well as written, oral, and presentation communication skills that are critically important for success in their professional careers.

Course Outcomes (CO):

On successful completion of this module, students should be able to learn:

1. How to write professional letter etc.
2. Public speaking

Course Contents:**Unit I: Elements of Communication**

Definition and Meaning of communication, Process of Communication, Essential components of the Process of Communication, Importance and Objectives of Communication, Differences between general and technical communication

Unit II: Types of Communication

Extra personal communication, Intrapersonal communication, Interpersonal communication, Organizational communication, Mass communication

Unit III: Verbal and Non-Verbal Communication

Verbal communication, Oral Communication, Advantages of Oral Communication, limitation/Disadvantages of Oral Communication

Non-verbal communication, Body Languages, Sign Languages, Space Languages, Paralanguages, Time Languages, Haptics or Languages of Touch, Color Languages, difference between verbal and non-verbal communication.

Unit IV: Formal and Informal Channels of Communication

Down ward Communication, Upward Communication, Horizontal /Lateral Communication, Diagonal /Clockwise Communication, and Merits & demerits of each type of communication.

Unit V: Barriers to Communication

Physical Barrier, Semantic/language Barrier, Sociopsychological Barrier, Organizational/hierarchical Barrier, Emotional Barrier, Cultural Barrier, Information overload,

Poor listening, Wrong assumption, Selective perception, Methods to overcome barriers to Communication, Qualities of Good Communication

Unit VI: Composition

1. Need and function of Business Letter, planning and layout of Business Letter, kinds of Business Letter 2. Drafting of business Letter: Sales, Credit, Enquiry, Order, Claim, Complaint, Job Applications, etc.

Unit VII: Preparation of Notices & circular, Memo, Declaration, Telephone etiquettes, E-mail writing

TEXT BOOK:

1. Hari Mohan Prasad, "Objective English", Tata McGraw Hill
2. A.Amin, R.Eravelly & F.J.Ibrahim, "Grammar Builder Level", ISBN : 9780521744843

TITLE OF COURSE: BASIC MATHEMATICS-I

COURSE CODE: BCA202

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Introduction:

The goal of this course is for students to gain proficiency in calculus computations. In calculus, we use three main tools for analyzing and describing the behavior of functions: limits, derivatives, and integrals. Students will use these tools to solve application problems in a variety of settings ranging from physics and biology to business and economics.

Course Outcomes (CO):

The students will learn:

1. Find intervals where a function is concave up or concave down.
2. Find inflection points.
3. Use the second derivative test to find local extrema.
4. Evaluate definite integrals to find net area between a curve and the x-axis using the Fundamental Theorem of Calculus.
5. Use basic integration properties to solve graphical net area problems.
6. Find the equilibrium point of supply and demand.
7. Use definite integrals to calculate consumer and producer surplus.

Course Contents:**Module 1:Function (5L)**

Definition, domain and range of function, types of functions (into, onto, one to one), composite function.

Module 2:Limit (3L)

Definition, first principle, properties, and simple problems related to limit. Some standard limits.

Module 3: Continuity (4L)

Definition, continuity of sum, product, difference and quotient of two continuous functions, simple problems.

Module 4: Differentiation (10L)

Definition, differentiation of simple functions using first principle, differentiation of trigonometric functions and inverse circular functions, method of substitution, differentiation of product and quotient of functions, maxima and minima of a function of single variable.

Module 5: Integration (10L)

Definition, integration of simple functions using substitution, integration of trigonometric and inverse circular functions and related problems, integration by parts, integration of rational functions. Definite integral and their properties, simple problems. Reduction formula and simple problems.

Reference Books

- Mathematics for Class 12 by R D Sharma
- NCERT Mathematics for Class 12

TITLE OF COURSE: OBJECT ORIENTED PROGRAMMING USING C++

COURSE CODE: BCA203

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts of C language.

Introduction: This course provides a comprehensive introduction to understand the underlying principles, Techniques and approaches which constitute a coherent body of knowledge in C++.

Course Outcomes (CO):

The course presents basics of C++ programming including: Basics of C++ environment, DataRepresentation, Control structures, Functions, Arrays, Pointers, Strings, and Classes that aims to:

1. Understand object oriented programming and able to explain the difference between object
2. Oriented programming and procedural programming.
3. Be able to program using C++ features such as composition of objects, operator overloads,
4. Dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
5. Be able to build C++ classes using appropriate encapsulation and design principles.
6. Understand functions and parameter passing. Know how to achieve polymorphism at compile
7. and run time with the concept of function overloading, operator overloading, virtual function
8. Understand object-oriented design and programming and also be familiar with the concept of
9. constructor destructor, inheritance
10. Understand dynamic memory allocation and pointers.

Course Contents:

Module-1: Introduction to Object-oriented Programming concept

Procedure-oriented Programming, Object-oriented Programming Paradigm; Basic concepts of Object oriented programming, Benefit of OOPs.

Module-2: Beginning with C++

What is C++? Application of C++,A simple C++ program, An example with class; Structure of C++ program, tokens, keywords, identifiers and constants, data types, reference variables, scope resolution operator.

Module-3: Functions in C++

Main function, function prototyping, call by reference,return by reference, Inline functions and friend functions, virtual function, Concept of Function overloading

Module-4: Classes and Objects:

Specifying a class, defining member functions; A C++ program with class; Making an Outside Function inline; Static data members; static member functions; arrays of objects; Objects as function arguments.

Module 5: Constructors and Destructors:

Constructors, default Constructors; Multiple constructors in a class; parameterized constructor; copy constructor; Destructor

Module 6: Inheritance:

Defining Derived classes, single inheritance; multilevel inheritance, multiple inheritance; hierarchical, hybrid inheritance; virtual base classes, abstract classes; constructor in derived classes; Making a private member inheritable.

Module 7: Operator overloading:

Defining Operator overloading, rules for overloading operators; Overloading unary operators using member function; Overloading of unary operator with friend function; Overloading Binary operators using member function; Overloading Binary operators using friends, Examples; Type conversion.

Module 8: Polymorphism:

Concept of polymorphism, runtime polymorphism, compile time polymorphism; Pointers, Pointers to objects; this pointer; Function overloading with an example (Program); Function Overriding with a proper example; Virtual function; Pure Virtual function; Abstract class

Module 9: Exception Handling:

Introduction, Basics of Exception Handling; Exception Handling mechanism; Throwing and catching mechanism; Rethrowing an Exception.

Text Books

1. E. Balagurusamy; Object Oriented programming with C++; Tata McGraw Hill Education Pvt. Ltd.
2. Schildt, H., the Complete Reference C++, Tata McGraw Hill Education Pvt. Ltd.

References

1. Debasish Jana, C++ object oriented programming paradigm, PHI.
2. D. Ravichandran, Programming with C++, Tata McGraw Hill Education Pvt. Ltd.

TITLE OF COURSE: OBJECT ORIENTED PROGRAMMING USING C++ LAB

COURSE CODE: BCA293

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in C programming languages, and Basic mathematics.

Objectives:

The course presents basics of C++ programming including: Basics of C++ environment, Data representation, Control structures, Functions, Arrays, Pointers, Strings, and Classes that aims to:

1. Understand object oriented programming and able to explain the difference between object oriented programming and procedural programming.
2. Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
3. Be able to build C++ classes using appropriate encapsulation and design principles.

Course Outcomes (CO):

1. Be able to develop, design and implement simple computer programs.
2. Understand functions and parameter passing.
3. Be able to do numeric (algebraic) and string-based computation.
4. Understand object-oriented design and programming.
5. Understand dynamic memory allocation and pointers.
6. Be able to design, implement, and test relatively large C++ programs.

Course Contents:**Exercises that must be done in this course are listed below:**

Experiment1:-Write a C++ program to take 10 integer data from the user and find out the maximum minimum from that data.

Experiment2:-

Write a C++ program to generate the Fibonacci series by using class.

Experiment3:-

Write a program to calculate $1+x+x*x+x*x*x.....$ using loop.

Experiment4:-

Write a program in C++ to find the reverse of a number.

Experiment 5:-

A shop required storing information about each item. Information will be item code, price and available quantities. User (sales person) will store information about each item and can display information about each item. Model the above problem with OOP.

Experiment6:-

A cricket organization need to store information like name, number of innings, number of not out innings, total run scored and total wicket taken of each cricketer. After storing data, organization will analyze the data and want to come on the following conclusion: If a cricketer

plays more or equal inning and is batting average is more than 35 then recognize him as a “BATSMAN”. if a cricketer plays more or equal to 50 innings’ and if taken more than 49 wickets then recognize him as a”BOWLER”.if one satisfies both condition then he will be “ALL ROUNDER”. Organization needs to display each information about each cricketer. Model above problem using OOPs.

Experiment 7:-

Create a class will two private integer data member, initialize them with constructor. Now display datamembers with the help of function which is not a member of that class.

Experiment8:-

Write a C++program to display the concept of function with default argument.

Experiment9:-

Create class ‘fun’ with one private float data member. Initialize that data member with constructor. Similarly create another class magic with private data member. Initialize that data member with constructor. Now using friend function check data member of which class is greater.

Experiment 10:-

Create a class test with one private float data member initialize that data member with constructor similarly create another class testing with one private data member. Initialize that data member with constructor. Now using function swap the value of data member of the classes.

Experiment11:-

Write a C++ program to demonstrate the concept of single inheritance.

Experiment12:-

Write a C++ program to demonstrate the concept of multiple inheritance.

Experiment 13:-

Write a C++ program to demonstrate the concept of MULTILEVEL inheritance.

Experiment 14:-

Write a C++ program to demonstrate the concept of HYBRID inheritance.

Experiment15:-

An application needs to swap two integer and two float values using functions. Approach the above problem using functions with same name.

Experiment16:-

Write aprogram to calculate the number of objects created by your program.

Experiment 17:-

Write a C++ program to achieve the following thing. A class contains 3 data member of type integer. Use ++ and -- operator in a way so that whenever we use ++ with the object of, all data member will incremented by one. Similarly, --will work.

Experiment18:-

Write a program to add two complex number using operators overloading.

Experiment19:-

Write a C++ program to demonstrate the concept of Virtual Class.

Experiment20:-

Write a C++ program to show how Run Time Polymorphism is achieved in C++.

Text Books

1. Schildt, H., The Complete Reference C++, Tata McGraw Hill Education Pvt. Ltd.
2. E.Balagurusamy; Object Oriented programming with C++; Tata McGraw Hill Education Pvt. Ltd.

References

1. Debasish Jana, C++ object oriented programming paradigm, PHI.
2. D. Ravichandran, Programming with C++, Tata McGraw Hill Education Pvt. Ltd.
3. Y.I. Shah and M.H. Thaker, Programming In C++, ISTE/EXCEL BOOKS.

TITLE OF COURSE: INTRODUCTION TO LINUX

COURSE CODE: BCA204A

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Linux, programming of shell script, basic of operating system task.

Introduction:

Most high-throughput bioinformatics tasks these days take place on the Unix environment and the most important software are designed to work best when used with a command-line interface. Because the command line can be an intimidating environment, many biologists tend to avoid it and limit their work on available web-based tools or local installation of software with graphic user interface. This means that they miss out the opportunity to access e.g. cloud computing or advance supercomputing which might help them to perform much more interesting analyses. The

best way to learn how to work in Unix command line environment is to be challenged in real problem trying to resolve them using “Trial and error” method through a self-learning process. Indeed, the Unix users form a very large community and the majority of the solutions are already available online in the web. This course is, therefore, designed to stimulate self and peer-learning using a set of very well designed and realistic bioinformatics problems typically occurring in the life of a biologist. The student will have access to 15 problems posted in a dedicated github account and one month to resolve them.

Course Outcomes (CO):

1. This course will serve to broaden the student's understanding of the issues and latest developments in the area of Linux. To reach this goal, the following objectives need to be met:
2. The Unix command line environment and to find online solutions for handling basic bioinformatics problems.
3. Identify and use UNIX/Linux utilities to create and manage simple file processing operations, organize directory structures with appropriate security, and develop shell scripts to perform more complex tasks.
4. Effectively use the UNIX/Linux system to accomplish typical personal, office, technical, and software development tasks.
5. Monitor system performance and network activities.
6. Effectively use software development tools including libraries, preprocessors, compilers, linkers, and make files.
7. Comprehend technical documentation, prepare simple readable user documentation and adhere to style guidelines.
8. Collaborate in teams on system tasks.

Course Contents:

Module 1: Introduction

The operating system: Linux history, Linux distribution, Linux feature, Linux relationship to Unix, overview of Linux architecture.

Module 2: Basic Commands

Basic Linux commands: date, cal, cd, cd.., cp, mv, cat, ls, ps, head, tail, wc, rm, rmdir, sort etc.

Module 3: vi editor

vi editor. General startup of vi editor and its modes, creating and editing files, features of vi screen, movement, cursor movement insertion, deletion searching, submitting operations, yank put, delete, commands reading & writing files.

Module 4: Basic of system Administrations

Installation, booting, login and shutdown process, start up scripts, controlling processes, system processes

Managing users: types of users, the powers of root, adding and deleting using the command line, introduction to Linux security.

Module 5: File System

The Linux file system: basic principles, pathnames, mounting and un-mounting file systems, different file types, file permissions, disk usage limits, directory structure, check and repair file system security and file permissions.

Module 6: Filter & Shell

Filter the grep family, advanced filters-sed and awk, meaning and purpose of shell, different Shells in Linux, the command line, standard input and output, redirection pipes, filter special characters for searching files and pathnames.

Module 7: Shell programming

Shell programming shell meta character local and global shell variables- interactive shell scripts – shell, script arguments – looping and making choice – for loop, case, while and until, shell functions level.

Module 8: System Call

Purpose of system call, Different system calls: File related system calls, Process related system calls, `Signal Handling using system calls.

Module 8: Inter-process communications

Inter-process communication: Pipes, shared memory, messages passing.

Module 9: Multithreading & Synchronization

Thread programming, Concurrency problem solve using mutex and semaphore.

Text Books

1. UNIX, concepts and applications by Sumitabha Das
2. Linux (2018 NEW Easy User Manual to Learn the Linux Operating System and Command Line by Yourself) by Brian Jones

References

1. Mastering Modern Linux by Paul S. Wang
2. Silberschatz, Galvin and Gagne, “Operating System Concepts”, 9th Edition, John Wiley & Sons Inc, 2013.
3. Andrew S. Tanenbaum, “Modern Operating Systems”, 4th Edition, Prentice-Hall of India, 2015.
4. SibsankarHaldar, Alex A.Aravind, “Operating systems”, Pearson Education, 2009

TITLE OF COURSE: INTRODUCTION TO LINUX LAB

COURSE CODE: BCA294A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students must have already registered for the course, basic programming concept.

Objectives: To study and apply concepts relating to operating systems, such as concurrency and control of asynchronous processes, deadlocks, memory management, processor and disk scheduling, parallel processing, and file system organization

Course Outcomes (CO):

Students will also be able to configure Linux system and troubleshoot and the differences between processes and threads, different process or thread synchronization methods and the tradeoffs between them, different memory management techniques used in Operating Systems, different I/O management techniques used in Operating Systems, tradeoffs in design and implementation concepts used in the development of Operating Systems

Course Contents:

Module 1: Introduction

The operating system: Linux history, Linux distribution, Linux feature, Linux relationship to Unix, overview of Linux architecture.

Module 2: Basic Commands

Basic Linux commands: date, cal, cd, cd.., cp, mv, cat, ls, ps, head, tail, wc, rm, rmdir, sort etc.

Module 3: vi editor

vi editor. General startup of vi editor and its modes, creating and editing files, features of vi screen, movement, cursor movement insertion, deletion searching, submitting operations, yank put, delete, commands reading & writing files.

Module 4: Basic of system Administrations

Installation, booting, login and shutdown process, start up scripts, controlling processes, system processes

Managing users: types of users, the powers of root, adding and deleting using the command line, introduction to Linux security.

Module 5: File System

The Linux file system: basic principles, pathnames, mounting and un-mounting file systems, different file types, file permissions, disk usage limits, directory structure, check and repair file system security and file permissions.

Module 6: Filter & Shell

Filter Thegrep family, advanced filters-sed and awk, meaning and purpose of shell, different Shells in Linux, the command line, standard input and output, redirection pipes, filter special characters for searching files and pathnames.

Module 7: Shell programming

Shell programming shell meta character local and global shell variables- interactive shell scripts – shell, script arguments – looping and making choice – for loop, case, while and until, shell functions eval.

Module 8: System Call

Purpose of system call, Different system calls: File related system calls, Process related system calls, `Signal Handling using system calls.

Module 8: Inter-process communications

Inter-process communication: Pipes, shared memory, messages passing.

Module 9: Multithreading & Synchronization

Thread programming, Concurrency problem solve using mutex and semaphore.

List Experiments

Experiment 1: Write a command to display calendar of November' 2020

Experiment 2: Write a command to display all the files and directories of present directory in long list format.

Experiment 3: Write a command to display first 4 lines and last 5 lines of a file.

Experiment 4: Write a command to remove a directory containing files and directories.

Experiment 5: Write a shell script to add two numbers.

Experiment 6: Write a shell script to swap two numbers using third variable.

Experiment 7: Write a shell script to swap two numbers without using third variable.

Experiment 8: Write a shell script to find the smallest numbers between two numbers.

Experiment 9: Write a shell script to find the smallest among three numbers.

Experiment 10: Write a shell script to find median among three numbers.

Experiment 11: Write a shell script to find factors of a number.

Experiment 12: Write a shell script to find common factors between two numbers.

Experiment 13: Write a shell script to find gcd of two numbers.

Experiment 14: Write a shell script function to sort elements of an array.

Experiment 15. Write a shell script function to concatenate two arrays.
Experiment 16. Write a shell script function to reverse an array.
Experiment 17: Write down some file related system call.
Experiment 18: Write a C program to implement process related system call.
Experiment 19: Write a C program to show PPID and PID of two related processes.
Experiment 20: Write a C program to handle of different signals.
Experiment 21: Write down name of signals which can't be handle.
Experiment 22: Write a C program to implement IPC using pipes.
Experiment 22: Write a C program to implement IPC using shared memory.
Experiment23: Write a C program to implement IPC using message passing.
Experiment 24: Write a C program to implement concept of multithreading.

References

1. UNIX, concepts and applications by Sumitabha Das
2. Linux (2018 NEW Easy User Manual to Learn the Linux Operating System and Command Line by Yourself) by Brian Jones
3. Mastering Modern Linux by Paul S. Wang
4. Silberschatz, Galvin and Gagne, "Operating System Concepts", 9th Edition, John Wiley & Sons Inc, 2013.
5. Andrew S. Tanenbaum, "Modern Operating Systems", 4th Edition, Prentice-Hall of India, 2015.
6. SibsankarHaldar, Alex A.Aravind, "Operating systems", Pearson Education, 2009

TITLE OF COURSE: COMPUTER FUNDAMENTALS & EMERGING TECHNOLOGY

COURSE CODE: BCA204B

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Fundamental Technology Concepts and Emerging Technologies.

Introduction:

Computers are so widely used in our day-to-day lives that imagining a life without them has become almost impossible. Learning computer fundamentals is a stepping stone to having an insight into how these machines work. Once the student is aware of the basic terminology that is

commonly used in computer science, he/she can then go on to develop useful computer programs that may help solve a user's problem. To provide an understanding of Fundamental Technology Concepts and Emerging Technologies in Computer Field. Includes practical skills in data capture, text editing with report formatting, effective presentation tools, efficient search techniques and online collaboration tools.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest Fundamental Technology Concepts and Emerging Technologies and Basic about the area of software development and maintenance. To reach this goal, the following objectives need to be met:

1. Learn Basic of Computer fundamental.
2. Learn how to convert data in computing language.
3. Learn data types and control structures of C
4. Learn to map problems to programming features of C.
5. Learn Fundamental Technology Concepts and Emerging Technologies.
6. Learn about Computing Trends in Internet.

Course Contents:

Module-1: Introduction to computer system, uses, types:

Generations, Classifications, Applications, Basic Organization. Input and output devices. Basic concept of computer memory, computer software and networks.

Module-2: Data Representation:

Number system and coding schemes (ASCII and UNICODE). Human Computer Interface: Relationship between Hardware and Software, Types of software, Operating system as user interface, utility programs. Role of Computers in: Business, Manufacturing, Mobile Computing, Public Sector, Media, Defense Services.

Module-3: Computing Language:

Introduction, Data types, Keywords, Variables, Operators, Assembler and compiler, Simple programs for C Language.

Module-4: Overview of Emerging Technologies:

Cloud Computing: Meaning Features, & Service models – Infrastructure as a service, Advantages and disadvantages, Mobile Computing: Meaning, Business Applications of Mobile computing, Virtual reality & Augmented Reality: Meaning and applications, IOT - Internet of Things: Meaning & Application.

Module-5: Computing Trends in Internet, Education and Research:

Internet-role and importance, Web Server and Web clients like web browser or web app, IP addressing : Public Vs Private, Static Vs Dynamic, world wide web and related protocols, e-Library, Google Scholar.

Text Books

1. Computer Fundamentals by A. Goel, Pearson Education, 2010.
2. Introduction to Information Technology by P. Aksoy and L. DeNardis, Cengage Learning, 2006
3. Computer Fundamentals fourth edition by Pradeep K. Sinha and Priti Sinha BPB publications

References

1. Introduction to Information Technology by ITL Education Solutions Limited, second edition.
2. O' Level made simple "introduction to ICT resources" by Satish Jain, Shashank Jain, Shashi Singh & M. Geethalyer, BPB publication.
3. Information Technology The breaking wave by Dennis Curtin Tata McGraw-hill edition

TITLE OF COURSE: COMPUTER FUNDAMENTALS & EMERGING TECHNOLOGY LAB

COURSE CODE: BCA294B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Fundamental Technology Concepts and Emerging Technologies.

Introduction:

Computers are so widely used in our day-to-day lives that imagining a life without them has become almost impossible. Learning computer fundamentals is a stepping stone to having an insight into how these machines work. Once the student is aware of the basic terminology that is commonly used in computer science, he/she can then go on to develop useful computer programs that may help solve a user's problem. To provide an understanding of Fundamental Technology Concepts and Emerging Technologies in Computer Field. Includes practical skills in data capture, text editing with report formatting, effective presentation tools, efficient search techniques and online collaboration tools.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest Fundamental Technology Concepts and Emerging Technologies and Basic about the area of software development and maintenance. To reach this goal, the following objectives need to be met:

1. Learn Basic of Computer fundamental.
2. Learn how to convert data in computing language.
3. Learn data types and control structures of C
4. Learn to map problems to programming features of C.
5. Learn Fundamental Technology Concepts and Emerging Technologies.
6. Learn about Computing Trends in Internet.

List of Experiments

Lab1.

Features of MS Windows based OS and any of the Linux flavor, setting up users and User rights in a computer, adding a printer, Software Installation Troubleshooting basic computer connections.

Lab 2.1:

- a) Write a program to swap two numbers.
- b) Write a program to find the greatest of three numbers and print the numbers in ascending order.

Lab 2.2: Report Formatting using Word Processing (Practical)

- a) Draft an official letter for job interview invitation/ job appointment/ invitation to an event, use mail merge to input the recipients list linking with database.
- b) Given a project report in PDF format transfer to word processor software and format to include title page, specified Paragraph and Page Formatting (page size, orientation, line spacing, font type and font size, Indent, bullets, paragraph formatting) details, Acknowledgement page, Table of contents page, List of figures page, List of Tables page, bibliography, references, distinct headers for each chapter, page numbering in roman for initial pages and normal from first chapter. The document should be checked for spelling errors and corrected appropriately.
- c) Design a certificate in landscape orientation with a border around the document. d. Design a Garage Sale sign.

Lab 2.3: Content Presentation using Presentation Software (Practical)

- a) Preparing presentation in areas such as Impact of Social Media on Youth, Emerging trends in Mobile Technology include appropriate slide animation, slide transitions, sound recording, slide timings, customer feedback video.
- b) Export the presentation as video or save as slide show.
- c) Prepare handouts for audience.

Lab 3: Web Applications (Practical)

- a) Scheduling tasks in Google Calendar
- b) Create/Upload documents / spreadsheets and presentations online.
- c) Share and Collaborate in real time

- d) d. safely store and organize your work of Google Drive or One Drive.

Lab 4: Internet Applications (Practical)

- a) Surfing the Internet, Using Email and Search Engines
- b) Advanced web search and translation services, Web search, image search, Search only for pages that contain (ALL the search terms contain the exact phrase you type, contain at least one of the words you type, do NOT contain any of the words you type, written in a certain language, created in a certain file format like ppt, pdf, rtf, doc, xls)
- c) Advanced search operators: Include search (“+” search), synonym search, OR search, Domain search, Numrange search, other advanced search features (Google, Local language, Technology Search, Date, Occurrences, Domains, Safe search),

Reference Books:

- 1) Introduction to Information Technology by ITL Education Solutions Limited, second edition.
- 2) „ O” Level made simple “introduction to ICT resources” by Satish Jain, Shashank Jain, Shashi Singh & M. GeethaIyer, BPB publication.
- 3) Computer Fundamentals fourth edition by Pradeep K. Sinha and Priti Sinha BPB publications
- 4) Information Technology The breaking wave by Dennis Curtin Tata McGraw-hill edition.
- 5) Computer Fundamentals by A. Goel, Pearson Education, 2010.
- 6) Introduction to Information Technology by P. Aksoy and L. DeNardis, Cengage Learning, 2006

TITLE OF COURSE: SIMULATION &MODELING

COURSE CODE: BCA205A

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Pre-requisite: Statistics and Probability, Automatic Control, Basic Technical English.

Introduction:

The course will introduce the basic concepts of computation through modeling and simulation that are increasingly being used by architects, planners, and engineers to shorten design cycles, innovate new products, and evaluate designs and simulate the impacts of alternative approaches. Students will use MATLAB to explore a range of programming and modeling concepts while acquiring those skills. They will then undertake a final project that analyzes one of a variety of scientific problems by designing a representative model, implementing the model, completing a verification and validation process of the model, reporting on the model in oral and written form, and changing the model to reflect corrections, improvements and enhancements.

Course Outcomes (CO):

The course aims at providing knowledge of methods and principles for building mathematical models for dynamic systems. Model construction from physical principles as well as from measurement data is covered. Properties of models are also studied by means of simulation. After completed course the student should be able to reach the following objectives:

1. Knowledge and understanding:

1. Understand different methods for random number generation.
2. Have a clear understanding of the need for the development process to initiate the real problem.
3. Have a clear understanding of principle and techniques of simulation methods informed by research direction.

2. Cognitive skills (thinking and analysis):

4. Be able to describe the components of continuous and discrete systems and simulate them - Be able to model any system from different fields.
5. Be able to implement numerical algorithm to meet simple requirements, expressed in English - Be able to discuss the simulation methods and select the suitable technique on the problems.

3. Communication skills (personal and academic):

6. Know how to simulate any discrete system using queuing systems.
7. Be able to work effectively with others

Course Contents (CO):

Module-1: Introduction to Simulation: Introduction Simulation Terminologies, Application areas, Model Classification, Types of Simulation, Steps in a Simulation study, Concepts in Discrete Event Simulation - Simulation Examples

Module-2: Mathematical models: Statistical Models Concepts, Discrete Distribution-Continuous Distribution, Poisson Process, Empirical Distributions, Queuing Models: Characteristics, Notation, Queuing Systems, Markovian Models, Generation of Pseudo Random numbers- Properties of random Techniques for generating random numbers ,Testing random number generators, Generating Random-Variates, Inverse Transform technique: Acceptance, Rejection technique , Composition & Convolution Method.

Module-3: Analysis of Simulation Data: Input Modeling, Data collection, Assessing sample independence, Hypothesizing, distribution family with data, Parameter Estimation, Goodness-of-fit tests, Selecting input models in absence of data, Output analysis for a Single system - Terminating Simulations, Steady state

Module-4: Verification and Validation, Model Building, Verification of Simulation Models, Calibration and Validation of Models, Validation of Model Assumptions, Validating Input Output Transformations

Module-5: Simulation of Computer Systems and Case Studies, Simulation Tools, Model Input, High level computer system simulation, CPU Memory Simulation, Comparison of systems via simulation, Simulation Programming techniques, Development of Simulation models.

Text Books

1. Jerry Banks and John Carson, “Discrete Event System Simulation”, Fourth Edition, PHI, 2005.
2. Geoffrey Gordon, “System Simulation”, Second Edition, PHI, 2006
3. Frank L. Severance, “System Modeling and Simulation”, Wiley, 2001.
4. Averill M. Law and W. David Kelton, “Simulation Modeling and Analysis, Third Edition, McGraw Hill, 2006.
5. Jerry Banks, “Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice”, Wiley, 1998.

Reference Books

1. Sheldon M. Ross: Introduction to Probability Models 7th Edition, Academic Press, 2002
2. Donald E. Knuth: The Art of Computer Programming - Volume 2: Semi Numerical Algorithms, 2nd Edition, PEARSON
3. Education, Reading MA, USA 2000
4. Sheldon M. Ross: Simulation 3rd Edition, Academic Press, 2002
5. M. Law and W. D. Kelton. Simulation Modeling and Analysis, 3rd Edition, McGrawHill, New York, USA, 1998

TITLE OF COURSE: SIMULATION &MODELING LAB

COURSE CODE: BCA295A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students will be required to purchase the student version of MATLAB and complete programming and modelling assignments.

Objectives:

Students will be capable to demonstrate basic programming skills – functions, arrays, loops, conditional statements; procedures

Course Outcomes: At the end of the course, the student should be able to understand the basic concepts of computation through modelling and simulation that are increasingly being used by architects, planners, and engineers to shorten design cycles, innovate new products, and evaluate designs and simulate the impacts of alternative approaches

1. Use a range of commercial software packages to construct, verify and validate models of the given systems
2. Define, describe and apply basic concepts related to modeling, identification and simulation.
3. Simplify a given model using static relations, substitution of variables using constants, neglect of small effects and aggregation of states.
4. Using conservation laws and constitutive relationships and other physical relations to model mechanical (in one dimension), electrical and flow systems, and combinations of these, in DAE form and (if possible) in state space representation.
5. Construct bond graphs for the type of systems mentioned above, simplify and analyze the bond graph according to causality conflicts, and from a given bond graph without conflicts make a state space representation.
6. Calculate the index of a given DAE.
7. Model and simulate mechanical (in one dimension) and electrical systems using the computer tools Simulink and 20Sim.
8. Identify a model of a real system by choosing experiment setup, post processing of data, model structure and validation.
9. Calculate asymptotic bias and variance properties of a given linear identification problem.
10. Decide if a given simulation method is implicit or explicit and how many steps it consists of, and calculation of the local and global error and stability area for simple simulation methods.
11. Oral and written communication Technical English

Course Contents:

Module-1: Introduction to Simulation: Introduction Simulation Terminologies, Application areas, Model Classification, Types of Simulation, Steps in a Simulation study, Concepts in Discrete Event Simulation - Simulation Examples

Module-2: Mathematical models: Statistical Models Concepts, Discrete Distribution-Continuous Distribution, Poisson Process, Empirical Distributions,

Module-3: Analysis of Simulation Data: Input Modelling, Data collection, Assessing sample independence, Hypothesizing, distribution family with data, Parameter Estimation

Module-4: Verification and Validation, Model Building, Verification of Simulation Models, Calibration and Validation of Models, Validation of Model Assumptions, Validating Input Output Transformations

Module-5: Simulation of Computer Systems and Case Studies, Simulation Tools, Model Input, High level computer system simulation, CPU Memory Simulation, Comparison of systems via simulation, Simulation Programming techniques, Development of Simulation models.

List of Experiments

1. Computer Generation of Random Numbers.
2. Chi-square goodness-of-fit test.
3. One-sample Kolmogorov-Smirnov test
4. Test for Standard Normal Distribution
5. Testing Random Number Generators.
6. Monte-Carlo Simulation.
7. Simulation of Single Server Queuing System.
8. Simulation of Two-Server Queuing System.
9. Simulate and control a conveyor belt system
10. Two-sample Kolmogorov-Smirnov test.

References

1. Sheldon M. Ross: Introduction to Probability Models 7th Edition, Academic Press, 2002
2. Donald E. Knuth: The Art of Computer Programming - Volume 2: Semi Numerical Algorithms, 2nd Edition, PEARSON Education, Reading MA, USA 2000
3. Sheldon M. Ross: Simulation 3rd Edition, Academic Press, 2002
4. M. Law and W. D. Kelton. Simulation Modeling and Analysis, 3rd Edition, McGrawHill, New York, USA, 1998

TITLE OF COURSE: IT & SOFTWARE TESTING

COURSE CODE: BCA205B

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Pre-requisite: Familiarity with system design and analysis(SAD), software engineering and Internet technology

Introduction:

This course presents a comprehensive study of software testing and quality control concepts, principles, methodologies, management strategies and techniques. The emphasis is here on understanding software testing process, planning strategy, criteria, and testing methods, as well as software quality assurance concepts & control process.

Course Outcomes (CO):

The Objective of this course is to enable a clear understanding and knowledge of the foundations, techniques, and tools in the area of software testing and its practice in the industry. The course will prepare students to be leaders in software testing. Whether you are a developer or a tester, you must test software. We can learn strengths and weaknesses of a variety of software testing techniques. To reach this goal, the following objectives need to be met:

1. After completion of this course the student will be able to plan, develop, and execute an automated test plan.
2. To understand fundamentals of software testing and testing techniques.
3. To learn various metrics of software quality.
4. To acquire knowledge on software quality management.
5. Able to test the software using various testing techniques.
6. Propose and defend innovative solutions to software quality assurance in the context of various software development environments.
7. To know about various ISO 9000 quality standards.
8. Understand basic concepts and terminology of internet technology.
9. Have a basic understanding of HTML, XML, CSS, WWW.
10. Be able to identify issues related to information security.

Course Contents:

Module-1: Introduction: Purpose, Productivity and Quality in Software, Testing Vs Debugging, Model for Testing, Bugs, Types of Bugs, Testing and Design Style.

Module-2: Flow/Graphs and Path Testing, Achievable paths, Path instrumentation, Application, Transaction Flow Testing Techniques.

Module-3: Data Flow Testing Strategies, Domain Testing: Domains and Paths, Domains and Interface Testing.

Module-4: Linguistic, Metrics, Structural Metric, Path Products and Path Expressions, Syntax Testing, Formats, Test Cases.

Module-4: Logic Based Testing, Decision Tables, Transition Testing, States, State Graph, State Testing.

Module-5: Introduction of IT: World Wide Web(WWW), HTML, XML, CSS.

Text Books

1. B. Beizer , 2003, Software Testing Techniques, II Edn., DreamTech India, New Delhi.
2. K.V.KK. Prasad , 2005, Software Testing Tools, DreamTech. India, New Delhi.
3. Software Testing techniques –BarisBeizermDreamtech, Second edition. 2
4. Software Testing Tools – Dr. K.V.K.K. Prasad, Dreamtech.
5. Software Testing Principles and Practices by Naresh Chauhan, Oxford University Press
6. Web Technology: A Developer's Perspective, N.P. Gopalan and J. Akilandeswari, PHI Learning, Delhi, 2013. (Chapters 1-5,7,8,9).
7. Internetworking Technologies, an Engineering Perspective, Rahul Banerjee, PHI Learning, Delhi, 2011. (Chapters 5,6,12)

Reference Books

- 1 I. Burnstein, 2003, Practical Software Testing, Springer International Edn.
2. E. Kit, 1995, Software Testing in the Real World: Improving the Process, PearsonEducation, Delhi.
3. R.Rajani, and P.P.Oak, 2004, Software Testing, Tata Mcgraw Hill, New Delhi.

TITLE OF COURSE: IT & SOFTWARE TESTING LAB**COURSE CODE: BCA295B****L-T-P: 0-0-3****CREDITS: 1.5**

SYLLABUS

Prerequisite: Familiarity with system design and analysis (SAD), software engineering and Internet technology

Objectives:

Students will be capable to demonstrate basic programming skills – functions, arrays, loops, conditional statements; procedures

Course Outcomes (CO):

The Objective of this course is to enable a clear understanding and knowledge of the foundations, techniques, and tools in the area of software testing and its practice in the industry.

Course Contents:

Module-1: Introduction: Purpose, Productivity and Quality in Software, Testing Vs Debugging, Model for Testing, Bugs, Types of Bugs, Testing and Design Style.

Module-2: Flow/Graphs and Path Testing, Achievable paths, Path instrumentation, Application, Transaction Flow Testing Techniques.

Module-3: Data Flow Testing Strategies, Domain Testing: Domains and Paths, Domains and Interface Testing.

Module-4: Linguistic, Metrics, Structural Metric, Path Products and Path Expressions, Syntax Testing, Formats, Test Cases.

Module-5: Logic Based Testing, Decision Tables, Transition Testing, States, State Graph, State Testing.

Module-6: Introduction of IT: World Wide Web(WWW), HTML, XML, CSS.

List of Experiments

1. Write Program in C language to demonstrate the following constructs:
do....while
while....do
if....else
switch
for
2. A program written in „C“ language for Matrix Multiplication fails! introspect the causes for its failure and write down the possible reasons for its failure.
3. Take any system (e.g. ATM system) and study its system specifications and report the various bugs.
4. Write the test cases for any known application (e.g. banking application)
5. Create a test plan document for any application (e.g. Library Management System)
6. Study of any testing tool (e.g. Win runner)
7. Study of any web testing tool (e.g. Selenium)
8. Study of any bug tracking tool (e.g. Bugzilla, bugbit)
9. Study of any test management tool (e.g. Test Director)
10. Study of any open source-testing tool (e.g. Test Link)

References

- 1 I. Burnstein, 2003, Practical Software Testing, Springer International Edn.
2. E. Kit, 1995, Software Testing in the Real World: Improving the Process, Pearson Education, Delhi.
3. R.Rajani, and P.P.Oak, 2004, Software Testing, Tata Mcgraw Hill, New Delhi.

TITLE OF COURSE: ENVIRONMENT & ECOLOGY

COURSE CODE: BCA206A

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Objectives:

The importance of Environmental Studies cannot be disputed. The need for sustainable development is a key to the future of mankind. The degradation of our environment is linked to continuing problems of pollution, loss of forest, solid waste disposal, issues related to economic productivity and national as well as ecological security. The increasing levels of global warming, the depletion of the ozone layer and a serious loss of biodiversity have also made everyone aware of growing environmental concerns. The United Nations Conference on Environment and Development held in Rio

De Janeiro in 1992, and the World Summit on Sustainable Development at Johannesburg in 2002 have drawn the attention of people around the globe to the deplorable condition of our environment. It is clear that no citizen of the earth can afford to be ignorant of environmental issues. Environmental management has become a part of the health care sector. Managing environmental hazards and preventing possible disasters has become an urgent need.

Human beings have been interested in ecology since the beginning of civilization. Even our ancient scriptures have included practices and values related with environmental conservation. It is now even more critical than ever before for mankind as a whole to have a clear understanding of environmental concerns and to follow sustainable development practices.

Course Outcomes (CO):

In spite of the deplorable status of the environment, the formal study of environment has so far not received adequate attention in our academic performances.

Course Contents:

Module-1: Multidisciplinary nature of environmental studies, Definition, scope and importance
Need for public awareness

Module-2: Renewable and non-renewable resources:

Natural resources and associated problems.

- a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people.

- b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems.
 - c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.
 - d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies.
 - e) Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies.
 - f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification.
- Role of an individual in conservation of natural resources.
 - Equitable use of resources for sustainable lifestyles..

Module-3: Ecosystems

- Concept of an ecosystem
- Structure and function of an ecosystem.
- Producers, consumers and decomposers.
- Energy flow in the ecosystem.
- Ecological succession.
- Food chains, food webs and ecological pyramids.
- Introduction, types, characteristic features, structure and function of the following ecosystem :-
- a. Forest ecosystem
- b. Grassland ecosystem
- c. Desert ecosystem
- d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Module-4: Biodiversity and its conservation

- Introduction – Definition: genetic, species and ecosystem diversity.
- Biogeographically classification of India
- Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values
- Biodiversity at global, National and local levels.

Module-5: Environmental Pollution

Definition, Cause, effects and control measures of:-

- a. Air pollution
- b. Water pollution
- c. Soil pollution
- d. Marine pollution
- e. Noise pollution
- f. Thermal pollution
- g. Nuclear hazards

Solid waste Management: Causes, effects and control measures of urban and industrial wastes, Role of an individual in prevention of pollution. Pollution case studies. Disaster management: floods, earthquake, cyclone and landslides.

Module-6: Social Issues and the Environment

- From Unsustainable to Sustainable development
- Urban problems related to energy
- Water conservation, rain water harvesting, watershed management
- Resettlement and rehabilitation of people; its problems and concerns. Case Studies
- Environmental ethics: Issues and possible solutions.
- Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies.
- Wasteland reclamation.
- Consumerism and waste products.
- Environment Protection Act.
- Air (Prevention and Control of Pollution) Act.
- Water (Prevention and control of Pollution) Act
- Wildlife Protection Act
- Forest Conservation Act
- Issues involved in enforcement of environmental legislation.
- Public awareness.

Module-7: Human Population and the Environment

- Population growth, variation among nations.
- Population explosion – Family Welfare Program.
- Environment and human health.
- Human Rights.
- Value Education.
- HIV/AIDS.
- Women and Child Welfare.
- Role of Information Technology in Environment and human health.
- Case Studies.

Text Books

1. Agarwal, K.C. 2001 Environmental Biology, Nidi Publ. Ltd. Bikaner.
2. Bharucha Erach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad – 380 013, India, Email: mapin@icenet.net (R)
3. Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p
4. Clark R.S., Marine Pollution, Clanderson Press Oxford (TB)
5. Cunningham, W.P. Cooper, T.H. Gorhani, E & Hepworth, M.T. 2001,
6. Environmental Encyclopedia, Jaico Publ. House, Mumabai, 1196p
7. De A.K., Environmental Chemistry, Wiley Eastern Ltd.

8. Down to Earth, Centre for Science and Environment (R)
9. Gleick, H.P. 1993. Water in crisis, Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute Oxford Univ. Press. 473p
10. Hawkins R.E., Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay (R)
11. Heywood, V.H & Waston, R.T. 1995. Global Biodiversity Assessment. Cambridge Univ. Press 1140p.

TITLE OF COURSE: MANAGEMENT ACCOUNTING

COURSE CODE: BCA206B

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Management Accounting.

Introduction:

This course provides the students an understanding of relevance of cost in managerial decision making. This course provides a comprehensive knowledge of classification of cost, apportionment of overheads, process costing, activity based costing, segmental reporting, preparation of budgets and cost -volume profit analysis for decision making and cost control.

Course Outcomes (CO):

At the end of the course, students are able to

1. Explain the concepts of unit costing activity based costing, apportionment of overheads, process costing, segmental reporting and budgeting.
2. Exhibit skills in Identifying, Measuring and analyzing costing data.
3. Provide alternative solutions to cost control and related cost management applications in practice.

Course Contents:

Module: 1

Background- Nature of Management Accounting

Financial Analysis- Cash Flow Statement (as per AS3), Financial Statements Analysis

Module: 2

Cost Accumulation- Fundamentals of Job-Order Batch & Process Costing, Variable Costing and Absorption (Full) Costing, Activity Based Costing System

Module: 3

Profit Planning- Cost -Volume-Profit Analysis, Budgeting and Profit Planning, Flexible Budgeting

Module: 4**Cost Control-**

Standard Costs and quality Costs, Cost Variance Analysis, Revenue and Profit Variance Analysis, Responsibility Accounting

Relevant Costing–Introduction–Relevant Costs and Revenues–Cost Concepts–

Outsourcing Decision– Decision to accept or reject a special order– Decision to continue or abandon a project.

Module: 5

Total Cost Management – Introduction – TCM and Business competitive edge - TCM Principles and implementation.

Text Books

1. Jiambalvo, James. (2004), Managerial Accounting, 2nd Edition, Wiley India Publications, New Delhi.

References

1. Khan, MY. Jain, P K (2000), Management Accounting, 3rd Edition; Tata McGraw Hill, New Delhi.
2. Jain, S P. Narang, K L. (2012), Cost Accounting: Principles and Practice, 23rd. Edition, Kalyani Publishers, Ludhiana.

TITLE OF COURSE: ESP & SDP - II**COURSE CODE: GSC201****L-T-P: 0-1-0****CREDITS: 1****Introduction:**

The course is on GK & CA, Quantitative Aptitude, Logical Reasoning and Objective English-2

Course contains:**Module-1****GK & CA**

Preamble, Fundamental rights & duties, parts & articles, DPSP, Panchayat, Municipality, MLA, MP, PM, President, Governor, SC, ST, UPSC, CAG of Indian constitution.

Basic economics- Types of Economy, Feature of Indian Economy, HDI

Sectors of the economy and their analysis: Primary (Agriculture, Mining, etc), Secondary (Industry, various policies), Tertiary (services, etc.) Liberalization, Privatization and Globalization (LPG), Imports and Exports

RBI & Its Function- Board of Governance, Operation.

Credit control policies- CRR, SLR, Bank rate, Repo rate, Reverse Repo rate, Prime lending rate, MSF, LAF, FERA, FEMA.

Union Budget & Railway Budget- concept of revenue, expenditure & different types of deficit.
Growth & Development of Economy, Sustainable development.
Current Affairs.

SKILL DEVELOPMENT FOR PROFESSIONALS II

Module-1

Quantitative Aptitude

- 1) Average- Concept on average, different missing numbers in average estimation, shortcuts & their application.
- 2) Mixture & Allegation – Proportion & mixtures in percentages, populations & liquids, shortcuts & their application.
- 3) Number System- concept of different numbers, remainder theorem, factors.
- 4) Time & Work- Basic concept, Different problems & their shortcut tricks. Time & Speed & Tides- concept of speed, time & Distance, relative speed, formulae & their application. Upstream & Downstream, Pipes & cistern.

Module-2

Logical Reasoning

- 1) Cube Dice, Miscellaneous Problems
- 2) Data Sufficiency
 - a) Problems on Blood Relation, ages, Numbers b) Logical Test Based on Data Sufficiency
- 3) Non Verbal Reasoning
 - a) Image Formation b) Water –Images
 - c) Mirror Image
 - d) Image completion
 - e) Paper Cutting and Folding

Module-3

Objective English-2

- 1) Clauses: Definition, Examples, Rules & Application, Types of Sentences
(Simple +Complex +Compound) Examples, Rules & Application, Voice- Concept, Types, Examples, Rules & Application, Narration Change- Rules (Direct & Indirect Speech)
- 2) Vocabulary- : Synonyms, Antonyms with examples, one word Substitution, Idioms & Phrases
- 3) Spotting Errors
- 4) Reading Comprehension (Level II)

Module-4 Data Interpretation level-II

Newspaper reading: The Hindu & Economic Times

Third Semester Syllabus							
Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	CC	BCA301	Operating System	3	0	3	4.5
2	CC	BCA302	Mathematics-II	3	0	0	3
3	CC	BCA303	Data Structures using Python	3	1	3	5.5
4	DE*		Elective-I	3	1	3	5.5
5	DE**		Elective-II	3	0	0	3
6	GE*		Generic Elective	3	0	0	3
7	GSC	GSC301	ESP & SDP-III	0	1	0	1
Total							25.5

Students have to take one mandatory NPTEL course-based on the respective year's offered courses.

Suggestive Choice Based Subjects

SI No	Type	Subject Code	Topic	L	T	P	Credit Points
1	DE*	BCA304A	Data Communication & Networking	3	1	3	5.5
2	DE*	BCA304B	Information Technology	3	1	3	5.5
3	DE**	BCA306A	Data Mining& Warehouse	3	0	0	3
4	DE**	BCA306B	System Analysis & Design	3	0	0	3
5	GE*	BCA305A	Cost Accounting	3	0	0	3
6	GE*	BCA305B	Business Accounting	3	0	0	3

TITLE OF COURSE: OPERATING SYSTEM

COURSE CODE: BCA301

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Introduction:

This course examines operating system design concepts, data structures and algorithms, and systems programming basics. The Topics to be covered (tentatively) include: Computer and operating system structures, Process and thread management, Process synchronization and communication, Memory management, Virtual memory, File system, I/O subsystem and device management, selected examples in networking, protection and security

Objectives:

In this course we will study the basic components of an operating system, their functions, mechanisms, policies and techniques used in their implementation and examples from popular operating systems. The way different modules in the operating system interact and work together to provide the basic services of an operating system.

Course Outcomes (CO):

Knowledge:

1. Understand the theory and logic behind the design and construction of operating systems.
2. You will examine the algorithms used for various operations on operating systems.
3. You will differentiate between various operating systems functionalities in terms of performance.
4. Become aware of the issues in the management of resources like processor, memory and input-output.
5. Know the problems in the design of operating system and study the probable solutions.
6. Learn to calculate the performance of cpu scheduling and disk scheduling
7. Learn File systems and methods of accessing
8. Understanding various security threats
9. An overview of advanced operating systems and compare the technical aspects of all the advanced operating systems

Application:

1. To develop, implement, and debug various CPU scheduling algorithms
2. To develop, implement, and demonstrate the algorithms of synchronizing the processes
3. To develop algorithms to find deadlocks
4. To develop Disk scheduling algorithms

Course Contents:

Unit 1: Introduction, Operating system structure - Monolithic systems, Layered systems, Virtual machines, Client-Server model.

Unit 2: Process Management – process creation, deletion, inter process communication tools: pipe, FIFO, shared memory, process synchronization, synchronization primitives and Classical IPC problems.

Unit 3: Process scheduling, Processor Allocation - Allocation Model, Design issues for processor allocation algorithms, Threads and Deadlock.

Unit 4: Memory Management, paging scheme, segmentation, virtual memory concept, page replacement algorithms, thrashing, working set model, issues in Virtual memory management.

Unit 5: File System management. Input output management, Disk scheduling, Case study of UNIX/LINUX.

Text Books

1. Silberschatz, P. Galvin and Greg Gagne, “Operating System Concepts”, Wiley International Company.
2. A.S. Tanenbaum, Modern Operating Systems, Prentice Hall India.

References

1. J. Archer Harris, Operating systems – Schuam’s outlines, Tata McGraw Hill.
2. Gary Nutt, Operating Systems – A modern perspective, Pearson Education.

TITLE OF COURSE: OPERATING SYSTEM LAB

COURSE CODE: BCA391

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS**Objectives:**

1. To learn and understand system calls related to files, processes, signals, semaphores and implement system programs based on that.
2. To provide an understanding of the design aspects of operating system.
3. To provide an efficient understanding of the language translation peculiarities by designing a complete translator for a mini language.

Course Outcomes (CO):

The students will have a detailed knowledge of the concepts of process and shared memory, aware of a variety of approaches to process management and main-memory management, including interference, deadlock, scheduling, fragmentation, thrashing, learn the basics behind file systems and input output systems and understand the fundamentals of network and distributed operating systems. Upon the completion of Operating Systems practical course, the student will be able to:

1. **Understand** and implement basic services and functionalities of the operating system using system calls.
2. **Use** modern operating system calls and synchronization libraries in software/ hardware interfaces.
3. **Understand** the benefits of thread over process and implement synchronized programs using multithreading concepts.
4. **Analyze** and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF, and Priority.
5. **Implement** memory management schemes and page replacement schemes.
6. **Simulate** file allocation and organization techniques.
7. **Understand** the concepts of deadlock in operating systems and implement them in multiprogramming system.

Course Contents:

Exercise No.1: CPU scheduling

Exercise No. 2: File allocation Strategy

Exercise No. 3: Simulate MVT, MFT(Multiprogramming Fixed and Variable)

Exercise No. 4: Simulate all File Organization Techniques

Exercise No. 5: Simulate Banker's Algorithm for Dead Lock Avoidance

Exercise No. 6: Simulate Banker's Algorithm for Dead Lock Prevention

Exercise No. 7: Simulate all page replacement Strategies

Exercise No. 8: Simulate Paging Technique of Memory Management

Exercise No. 9: Shell programming (cut, grep, sed)

Exercise No. 10: Process

Text Book:

1. Maurice J. Bach, Design of the UNIX Operating System, PHI.

Recommended Systems/Software Requirements:

1. Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
2. Turbo C or TC3 compiler in Windows XP or Linux Operating System.

TITLE OF COURSE: MATHEMATICS- II(BASIC STATISTICS AND NUMERICAL METHODS)

COURSE CODE: BCA302

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Introduction:

The undergrad curriculum in Statistics provides majors with the skills needed to utilize statistical techniques for addressing quantitative, data-based problems in fields such as biological and social sciences, engineering and technology, business and finance, law, and health and education

Course Outcomes:

The students will learn:

1. statistical reasoning and inferential methods
2. statistical modeling and its limitations
3. interpreting and communicating the results of a statistical analysis
4. data analysis using statistical computing tools and software
5. probability and the mathematical foundations of statistics
6. Knowledge and Understanding: Students are able to understand the nature and operations of Numerical Analysis.
7. Intellectual Skills: By the end of the course the student is expected to solve real-life and Engineering applications.

Course contents:

Statistics:

Basic Statistic (5L)

Scope, functions and limitations of statistics, Collection and presentation of data– Tabular and diagrammatic representation, Frequency distribution, relative frequency, cumulative frequency; Bar graphs and pie charts; Histogram.

Measures of Central tendency (3L)

Mean, Median, Mode, Percentiles, Quartiles

Measures of Dispersion (6L)

Range, Interquartile range, Mean deviation, Mean Absolute deviation, Standard deviation, Variance, Coefficient of Variation. Measures of shape and relative location; Skewness and Kurtosis, Simple correlation and regression analysis

Numerical Methods:

Finite Differences (4L)

Definition of operators and relations among them (All operators)

Interpolation (10L)

Interpolation With Equal Intervals: Newton's forward difference formula, Newton's backward difference formula. Interpolation with unequal intervals: Lagrange's Method, Newton divided Difference.

Solution of Algebraic and Transcendental Equations (6L)

Bisection method, Regula-Falsi method, Newton-Raphson Method, Iterative method.

Text Books:

- R. K. Jain, S.R.K. Iyengar & M.K. Jain, "Numerical Methods", New Age International Publishers, 2013.
- B. S. Grewal, "Numerical Methods", Khanna publications, 2014.
- S. C. Gupta, "Fundamentals of statistics", Sultan Chand & Sons, Delhi, 2006.
- Goon, Gupta and Das Gupta, "Fundamentals of statistics", The World Press Pvt. Ltd., Kolkata, 2008.

Reference Books:

- Mathews & Fink, "Numerical Methods using MATLAB", Pearson, 2010.
- Gerald and Wheatley, "Applied Numerical Analysis", Pearson, 2010
- Rohtagi, V.K., "An Introduction to Mathematical Statistics", John Wiley & Sons, 1976.
- Mood A.M., Greybill, F.A. and Bose D.C, "Introduction to the Theory of Statistics", McGraw Hill, 1974.
- S.P. Gupta, "Statistical Methods", Sultan Chand & Sons, New Delhi, 2008.

TITLE OF COURSE: DATA STRUCTURES USING PYTHON

COURSE CODE: BCA303

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Pre-requisite: Basic concepts in mathematics and programming languages.

Introduction:

This course examines data structures and algorithms basics using python. The Topics to be covered (tentatively) include: an introduction to programming and problem solving in Python with basic concepts such as conditionals, loops, functions, lists, strings and tuples; Time and space analysis of algorithms; Linear Data structures like array, linked list, stack, queue; Non-linear Data structures like graph and tree; Sorting; Searching and Hashing.

Course Outcomes (CO):

In this course we will study the basic components of data structure and algorithm. Students are expected to be capable of understanding the data structures, their advantages and drawbacks, how to implement them in python, how their drawbacks can be overcome and what the applications are and where they can be used. To reach this goal, the following objectives need to be met:

1. Know about basic concepts of python programming.
2. To learn about the data structures/ methods/algorithms mentioned in the course with a comparative perspective so as to make use of the most appropriate data structure/ method/algorithm in a program to enhance the efficiency (i.e. reduce the run-time) or for better memory utilization, based on the priority of the implementation
3. To implement different types of linear data structure.
4. Study the Python dictionaries as well as classes and objects for defining non linear data structure like graph and tree.
5. Understand different types of sorting, searching and hashing technique.

Course Contents:

Module-1: Basics of python: data types, assignment statements, control flow, strings, lists, functions, simple input output.

Module-2: Algorithm efficiency and analysis, time and space analysis of algorithms – order notations.

Module-3: Linear Data structures–Array, Matrix, Linked List, Stack, Queue and Recursion with their types, different operations and applications

Module-4: Nonlinear Data structures–Graph, Trees, Minimum spanning tree with their types, different operations and applications.

Module-5: Sorting, Searching and Hashing- Bubble sort and its optimizations, insertion sort, shell sort, selection sort, merge sort, quick sort, heap sort (concept of max heap, application – priority queue), radix sort. Sequential search and binary search. Hashing functions, collision resolution techniques

Text Books

1. Data Structures and Algorithms in Python: Michael H. Goldwasser, Roberto Tamassia, Michael T. Goodrich, Publisher: John Wiley & Sons
2. Data Structure and Algorithmic Thinking with Python: Narasimha Karumanchi; Careermonk publication.

References

1. Problem Solving in Data Structures & Algorithms Using Python: Programming Interview Guide: Hemant Jain; Createspace Independent Pub
2. Data Structures and Algorithms Using Python: Neceise Rance D; Wiley publisher

TITLE OF COURSE: DATA STRUCTURES USING PYTHON LAB

COURSE CODE: BCA393

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Basic concepts in mathematics and programming languages.

Introduction:

This course examines data structures and algorithms basics using python. The Topics to be covered (tentatively) include: an introduction to programming and problem solving in Python with basic concepts such as conditionals, loops, functions, lists, strings and tuples; Time and space analysis of algorithms; Linear Data structures like array, linked list, stack, queue; Non-linear Data structures like graph and tree; Sorting; Searching and Hashing.

Objectives:

1. Develop problem solving ability using Programming.
2. Develop ability to design and analyze algorithms.
3. Introduce students to data abstraction and fundamental data structures.
4. Develop ability to design and evaluate Abstract Data Types and data structures.
5. Apply data structure concepts to various examples and real life applications

Learning Outcomes:

The course will use hands on practice and applying the knowledge gained in theory course to different day to day real world applications. Upon the completion of data structure and algorithm practical course, the student will be able to:

1. **Understand** and implement different type of data structure techniques
2. **Analyze** the hashing method.
3. **Implement** different type sorting searching techniques.

Course Contents:

Exercises that must be done in this course are listed below:

Exercise No.1:Implementation of array operations

Exercise No. 2:Stacks and Queues: adding, deleting elements

Exercise No. 3:Circular Queue: Adding & deleting elements

Exercise No. 4:Merging Problem: Evaluation of expressions operations on multiple stacks & queues

Exercise No. 5:Implementation of linked lists: inserting, deleting, and inverting a linked list.

Exercise No. 6:Implementation of stacks & queues using linked lists, Polynomial addition, and Polynomial multiplication

Exercise No. 7:Sparse Matrices: Multiplication, addition.

Exercise No. 8:Recursive and Non-recursive traversal of Trees

Exercise No. 9:Threaded binary tree traversal. AVL tree implementation

Exercise No. 10:Application of Trees. Application of sorting and searching algorithms

Text Books

1. Data Structures and Algorithms in Python:Michael H. Goldwasser, Roberto Tamassia, Michael T. Goodrich,Publisher: John Wiley & Sons
2. Data Structure and Algorithmic Thinking with Python:NarasimhaKarumanchi; Careermonk publication.

References

1. Problem Solving in Data Structures & Algorithms Using Python: Programming Interview Guide:Hemant Jain; Createspace Independent Pub
2. Data Structures and Algorithms Using Python: NecaiseRance D; Wiley publisher

TITLE OF COURSE: DATA COMMUNICATION & NETWORKING**COURSE CODE: BCA304A****L-T-P: 3-1-3****CREDITS: 5.5**

SYLLABUS**Introduction:**

This course is to provide students with an overview of the concepts and fundamentals of data communication and computer networks. Topics to be covered include: data communication concepts and techniques in a layered network architecture, communications switching and routing, types of communication, network congestion, network topologies, network configuration and management, network model components, layered network models (OSI reference model, TCP/IP networking architecture) and their protocols, various types of networks (LAN, MAN, WAN and Wireless networks) and their protocols. The course is supplemented by a practical component covered in CS692 concurrently.

Course Outcomes (CO):

After completing this course the student must demonstrate the knowledge and ability to:

1. Independently understand basic computer network technology.
2. Understand and explain Data Communications System and its components.
3. Identify the different types of network topologies and protocols.
4. Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
5. Identify the different types of network devices and their functions within a network.
6. Understand and building the skills of subnetting and routing mechanisms.
6. Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.
7. 8. Analyse the features and operations of various application layer protocols such as Http, DNS, and SMTP.

Application:

1. To configure and implement network topology.
2. To configure and implement local area network.
3. To design network and assign IP address
4. Connect Remote computers
5. Analyze the network.

Course Contents:**Unit-1:**

Introduction; Data communications: components, data representation (ASCII,ISO etc.),direction of data flow(simplex, half duplex, full duplex); Networks: distributed processing, network criteria, physical structure (type of connection, topology), categories of network (LAN, MAN,WAN);Internet: brief history, internet today; Protocols and standards; Reference models: OSI reference model, TCP/IP reference model, their comparative study.

Physical layer:

Overview of data(analog & digital), signal(analog &digital), transmission (analog & digital)& transmission media (guided & non-guided); TDM, FDM, WDM; Circuit switching: time division & space division switch, TDM bus; Telephone network;

Unit-2:

Data link layer:

Types of errors, framing(character and bit stuffing), error detection & correction methods; Flow control; Protocols: Stop & wait ARQ, Go-Back- N ARQ, Selective repeat ARQ, HDLC;

Medium access sub layer:

Point to point protocol, LCP, NCP, FDDI, token bus,token ring; Reservation, polling, concentration; Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, FDMA, TDMA, CDMA; Traditional Ethernet, fast Ethernet;

Unit-3:

Network layer:

Internetworking & devices: Repeaters, Hubs, Bridges, Switches, Router, Gateway; Addressing : Internet address, classful address, subnetting; Routing : techniques,static vs. dynamic routing , routing table for classful address; Routing algorithms: shortest path algorithm, flooding, distance vector routing, link state routing; Protocols: ARP, RARP, IP, ICMP, IPV6; Unicast and multicast routing protocols.

Transport layer:

Process to process delivery; UDP; TCP; Congestion control algorithm: Leaky bucket algorithm, Token bucket algorithm, choke packets; Quality of service: techniques to improve QoS.

Unit-4:

Application layer:

DNS; SMTP, SNMP, FTP, HTTP & WWW; Security: Cryptography, user authentication, security protocols in internet, Firewalls.

Modern topics:

ISDN services & ATM ; DSL technology, Cable modem, SONET. Wireless LAN: IEEE 802.11; Introduction to blue-tooth, VLAN's, Cellular telephony & Satellite network.

Text Books:

1. B. A. Forouzan – “Data Communications and Networking (3rd Ed.)” – TMH
2. A. S. Tanenbaum – “Computer Networks (4th Ed.)” – Pearson Education/PHI
3. W. Stallings – “Data and Computer Communications (5th Ed.)” – PHI/ Pearson Education
4. Zheng & Akhtar, Network for Computer Scientists & Engineers, OUP
5. Black, Data & Computer Communication, PHI
6. Miller, data Communication & Network, Vikas

Reference Books:

1. Kurose and Rose – “Computernetworking -A top down approach featuring the internet” – Pearson Education
2. Leon, Garica, Widjaja – “Communication Networks” – TMH
3. Walrand – “Communication Networks” – TMH.
4. Comer – “Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.)” – Pearson Education/PHI

TITLE OF COURSE: DATA COMMUNICATION & NETWORKING LAB

COURSE CODE: BCA394A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Objectives:

This practical course provides students with hands on training regarding the design, troubleshooting, modeling and evaluation of computer networks. In this course, students are going to experiment in a real and simulation based test-bed networking environment, and learn about network design and troubleshooting topics and tools such as: network addressing, Address Resolution Protocol, basic troubleshooting tools (like ping, ICMP), IP routing (e.g. RIP), TCP and UDP, DHCP, ACL and many others. Student will have the opportunity to build some simple networking models using the tool and perform simulations that will help them evaluate their design approaches and expected network performance.

Course Outcomes (CO):

The students will have a detailed knowledge network topology, Local area network, IP addressing, familiarization with network simulator, idea about networking devices, network cable and connectors, different types routing protocols, concept of remote access and different types of application layer protocol. Upon the completion of Computer network practical course, the student will be able to:

1. **Learn** various network commands.
2. **Understand** and implement basic of Network and Network Topology.
3. **To get** idea about IP addressing schemes.
4. **Understand** the benefits of network.
5. **Configure** and simulate various protocols.
6. **Access** remote desktop.
7. **Connect** to different computer using LAN.
8. **Understand** the concepts of access control.

Course Contents:

Exercises that must be done in this course are listed below:

Exercise No.1: Study of different types of Network cables and practically implements the cross-wired cable and straight through cable using clamping tool.

Exercise No. 2: Familiarization with some network devices.

Exercise No. 3: Study of Network IP.

Exercise No. 4: Connect the computers in LAN.

Exercise No. 5: Introduction to Packet Tracer.

Exercise No. 6: Configure network topology using packet tracer.

Exercise No. 7: Configure network topology using packet tracer to find the routing path by IPRoute Command.

Exercise No. 8: Network Configuration using distance vector routing protocol.

Exercise No. 9: Configuration of DHCP Protocol

Exercise No. 10: Telnet Configuration.

Exercise No. 11: Configuration of Access Control List.

Text Book:

1. B. A. Forouzan – “Data Communications and Networking (3rd Ed.) “ – TMH

Reference Book:

1. Authorized Self-Study Guide “Interconnecting Cisco Network Devices, Part 1(ICND1), 2nd Edition, January, 2008.

Recommended Systems/Software Requirements:

1. CAT-5/CAT-6 Cables, RJ 45, Cutter, Clamping Tool, Router , Switch and Hub.

2. Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
3. Turbo C or TC3 compiler in Windows XP or Linux Operating System.

TITLE OF COURSE: INFORMATION TECHNOLOGY

COURSE CODE: BCA304B

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Pre-requisite: Preliminary knowledge of computer, their operations and applications.

Introduction:

The main objective is to introduce IT in a simple language to all undergraduate students, regardless of their specialization. It will help them to pursue specialized programs leading to technical and professional careers and certifications in the IT industry. The focus of the subject is on introducing skills relating to IT basics, computer applications, programming, interactivemedias, Internet basics etc.

Course Outcomes (CO):

This course provides students information technology. This course will also provide a hands-on experience on computer. To reach this goal, the following objectives need to be met:

1. Understand basic concepts and terminology of information technology.
2. Have a basic understanding of personal computers and their operations.
3. Be able to identify issues related to information security.
4. Classification of computers
5. Information technology
6. Application of computer / information technology
7. Parts of a computer:
 - Input unit - keyboard, Pointing Devices (Mouse, Trackball, Touch Panel, and Joystick), Light Pen, Scanners, web cam, Digitizers, MICR, OCR, OMR, Bar-code Reader.
 - Output-monitor-CRT and LCD, printers- Impact Printers (Daisy Wheel, Dot Matrix, Line Printer, Chain Printer), Non-Impact Printers (inkjet, Laser Printer, Barcode Printers, Electro static printers and plotters).
 - Storage- primary and secondary, Data Storage and Retrieval methods, Classifications Volatile Memory and Non- Volatile , ROM, RAM, EPROM, PROM, EEPROM, Cache Memory, magnetic disks, optical disks (CDROM, WORM, DVD,

Blue Ray Disc), Flash Memory, SD/MMC Memory cards, File system, File Allocation Table (FAT , FAT 32 & NTFS).

- CPU- ALU, CU, processor speed

Course Contents:

Module-1: What are computers? The evolution of computers, Classification of computers, The control unit, computer organization & Block diagram representation, Decimal number system, binary number system, conversion of a binary number to decimal number, conversion of a decimal number to a binary number, addition of binary number, binary subtraction, hexadecimal number system, octal number system

Module-2: Storage devices, Input-Output devices, Low level and high level languages, assemblers, compilers, interpreters, linkers, algorithms, flow charting, decision tables, pseudo code, software, application software packages

Module-3: Operating system concepts, Different types of operating systems, structure of operating system, DOS/UNIX/LINUX commands, Data Processing, File systems and Database Management Systems, different types of Database Management System.

Module-4: Basic elements of a Communication System, Data transmission modes, Data Transmission speed, Data transmission media, Digital and Analog Transmission, Network topologies, Network Types (LAN, WAN and MAN), Communication protocols, Inter networking tools, Distributed Computing Systems.

Text Books

1. Alex Leon & Mathews Leon, “Fundamentals of Information Technology”, Leon Techworld, 1999.
2. P. K. Sinha & Priti Sinha, “Computer Fundamentals”, BPB Publications, 1992.
3. V. Raja Raman, “Introduction to Computers”, PHI, 1998.

Reference Books

1. Alex Leon & Mathews Leon, “Introduction to Computers”, Vikas Publishing House, 1999.
2. Norton Peter, “Introduction to computers”, 4th Ed., TMH, 2001.
3. Simon Haykins, “Communication System”, John Wiley & Sons, 1999

TITLE OF COURSE: INFORMATION TECHNOLOGY LAB

COURSE CODE: BCA394B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Preliminary knowledge of computer, their operations and applications.

Objectives:

This course provides students information technology. This course will also provide a hands-on experience on computer

Course Outcomes (CO):

The Objective of this course is to enable a clear understanding and knowledge of the foundations, techniques, and tools in the area of software testing and its practice in the industry.

Course Contents:

Module-1: What are computers? The evolution of computers, Classification of computers, the control unit, computer organization & Block diagram representation.

Module-2: Storage devices, Input-Output devices, Low level and high level languages, assemblers, compilers, interpreters, linkers, algorithms, flow charting, decision tables, pseudo code, software, application software packages

Module-3: Operating system concepts, Different types of operating systems, structure of operating system, DOS/UNIX/LINUX commands,

Module-4: Basic elements of a Communication System, Data transmission modes, Data Transmission speed, Data transmission media

List of Experiments

1. Create And Manage Files And Folder Tree Use Accessories Utilities Of Windows OS
2. Entering and Editing Text in Document File.
3. Apply Formatting Features on Text like Bold, Italics, Underline, Font Type, Colour and Size.
4. Apply Features Like Bullet, Numbering
5. Create Documents, Insert Images, Format Tables
6. Create And Manipulate Tables
7. Entering And Editing Data In Worksheet Apply Formula And Functions In The Sheet
8. Use Graphics And Auto Shapes In Excel Sheet
9. Create And Manipulate Excel Charts
10. Create Pay Bills, Pay Slips, Electricity Bills Using Excel Print Sheet Using Print area
11. Basic Operations Of Power Point, Create PPT and Inset And Delete Slides
12. Create Project Presentations, Lecture Presentations.
13. Use Of Mater Slide In Presentation
14. Apply Basic Formatting Features In Presentation Like Font, Font Size, Font Colour, Text Fill, Spacing And Line Spacing Formatting Text Boxes, Word Arts, Styles Bullet and Numbering
15. Working With Drawing Tools, Applying• Shape Or Picture Styles, Applying Object Borders, Object Fill, Object Effects. Working With Video, Link to Video and Sound Files.
16. Creating Hyperlinks, Using Action Buttons

References

1. I. Burnstein, 2003, Practical Software Testing, Springer International Edn.
2. E. Kit, 1995, Software Testing in the Real World: Improving the Process, Pearson Education, Delhi.
3. R.Rajani, and P.P.Oak, 2004, Software Testing, Tata Mcgraw Hill, New Delhi.

TITLE OF COURSE: DATA MINING& WAREHOUSE

COURSE CODE: BCA306A

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and computer architecture.

Introduction:

The recent years have generated explosive expansion of digital data stored in computer databases as well as increased pressure on companies to keep competitive advantage. This has put Data Mining (DM) as a key method for extracting meaningful information from the flood of digital data collected by businesses, government, and scientific agencies.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of Data mining and Data Warehouse. To reach this goal, the following objectives need to be met:

1. To understand the basic principles, concepts and applications of data warehousing and data mining.
2. To introduce the task of data mining as an important phase of knowledge recovery process.
3. Ability to do Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment.
4. Have a good knowledge of the fundamental concepts that provide the foundation of data mining.

5. Design a data warehouse or data mart to present information needed by management in a form that issuable for management client.
6. Have application of Weather Forecasting, Disaster Forecasting, Business Management, Market Analysis and Management Corporate Analysis & Risk Management.
7. Idea of Big data and hadoop in different data set.

Course Contents:

Module-1: Overview of data mining process:Data Mining, Data Warehouse, KDD, Data Mart, Data Frequency

Module-2: Data Mining Processes and Knowledge DiscoveryData Cleaning Unit, Data Integration Unit, Mining

Module-3: Database Support to Data Mining, Real life application on data set, Data Mining Techniques and Functions, Knowledge grow and prediction

Module-4: Cluster Analysis: K-mean, K-Median, DBSCAN, CLARA, CLARANS, ROCKetcRegression Algorithms in Data Mining:Regression is a data mining technique used to predict a range of numeric values, given a particular dataset, Neural Networks in Data Mining GA Algorithmusing R language.

Module-5: Decision Tree Algorithms: F-P Tree, Decision Tree (Entropy)

Module-6: Data Mining for Customer Relationship Management Fraud detection, and risk management applications, Customer Relationship Management, Risk Management.

Module-7: Link Analysis in Text Mining, Web Mining Taxonomy, Mining the Web User Behavior, Web Analytics, Web Mining, Text Mining etc

Module-8:Introduction to Big Data Topics, Rise of Big Data, Compare Hadoop vs traditional systems, Limitations and Solutions of existing Data Analytics Architecture, Attributes of Big Data, Types of data, other technologies vs Big Data, Idea of using R Language.

Module-9:Hadoop Architecture and HDFS Topics - What is Hadoop? Hadoop History, Distributing Processing System, Core Components of Hadoop, HDFS Architecture, Hadoop Master – Slave Architecture, Daemon types - Learn Name node, Data node, Secondary Name node. Hadoop Clusters.

Text Books

- 1.Data Mining Technology, Third Edition by Arun K Pujari, Universities Press, India
2. Data Warehousing Fundamentals for IT Professionals, Second Edition by PaulrajPonniah, Wiley India
3. Alex Berson, Stephen J. Smith, “Data Warehousing Data Mining & OLAP”, Tata McGraw-HillReferences

References

1. Data Warehousing, Data Mining, & OLAP – Second Edition by Alex Berson and Stephen J. Smith, Tata McGraw Hill
2. Data warehouse Toolkit by Ralph Kimball, Wiley India
3. Gajendra Sharma, “Data Mining Data Warehousing and OLAP”, S.K. KATARIA & SONS.
4. Sam Anahory, Dennis Murray, “Data Warehousing in the Real World”, PEARSON

TITLE OF COURSE: SYSTEM ANALYSIS & DESIGN

COURSE CODE: BCA306B

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Familiarity with programming languages and knowledge of databases.

Introduction:

The course deals with planning the development of information systems. The course deals with concepts, methodologies, techniques and tools required for solving business problems.

Course Outcomes (CO):

The course will teach techniques in data requirements collection and analysis along with methods to model data needs. To reach this goal, the following objectives need to be met:

1. Be able to develop requirements document to detail and model an information system design.
2. Demonstrate ability to initiate, specify, and prioritize information systems projects and determine various aspects of feasibility of these projects.
3. Be able to gather data to analyze and specify the requirements of a system.
4. Apply knowledge of data flow diagrams, entity relationship modeling and state process modeling in user requirement analysis.
5. Be able to recognize and incorporate user requirements, business rules and constraints into the design of an information system.
6. Display awareness of the alternative methods for system development and tools and techniques used in system analysis and design.
7. Apply specific modeling methodology to analyze, design and specify requirements for a business situation using a formal technique.
8. Exhibit competence in applying contemporary analysis and design techniques in business process improvement.

Course Contents:

Module-1: Fundamentals of Data and Information: Types of information: operational, tactical, strategic and statutory, need for information systems, management structure, Information requirements at different levels of management, functional allocation of management, requirements of information for various functions, Information quality

Module-2: Systems Analysis and Design Life Cycle: Requirements Specifications, Feasibility analysis, Final specifications, System design & implementation, System evaluation & modification. Role of systems analyst, Tools used in system analysis.

Module-3: Feasibility analysis: Deciding project goals, examining alternative solutions, cost, benefit analysis, quantifications of costs and benefits, payback period, system proposal preparation for managements, Tools for prototype creation.

Module-4: Analysis Process: Data flow diagrams, case study for use of DFD, good conventions, leveling of DFDs, logical and physical DFDs, Describing process specifications and structured decision, software tools to create DFDs.

Module-5: Essentials Of System Design: Procedure specifications in structured English, Decision tables for complex logical specifications, specification oriented design vs procedure oriented design.

Module-6: Advanced Systems Design Concepts: Data oriented system design, ER model and diagram, relation normalization and need, Object oriented system design, classification of objects in an application, System modeling using objects.

Module-7: System analysis and design case study: E-commerce system architecture, physical networks, logical network, world wide web, web-services.

Text Books

1. J. W. Satzinger, R. B. Jackson and S. D. Burd – "Systems Analysis and Design in a Changing World" – 6th Ed. – Thomas Course Technology.
2. Kenneth E. Kendall, Julie E. Kendall – "Systems Analysis and Design" – 9th Ed. – Pearson.

References

1. Brijendra Singh – "Systems Analysis and Design" – New Age International Publishers
2. Jeffrey Whitten & Lonnie Bentley – "Systems Analysis and Design Methods" – SIE Publishers
3. Valacich, George, Hoffer – "Essentials Of System Analysis & Design" – PHI Publishers
4. Charles Wasson – "System Engineering Analysis, Design, and Development: Concepts, Principles, and Practices " – Wiley series
5. Silver & Silver- "System Analysis and Design" -5th Ed.- Addison Wesley.

TITLE OF COURSE: COST ACCOUNTING

COURSE CODE: BCA305A

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Accounting System And Basic of Accounting

Introduction: Cost accounting as a discipline in the business programmer equips students with the basics of cost awareness and cost reduction to ensure creation and cautious use of resources in small and large scale organizations. Sections selected include basic principles, costing methods for cost and pricing purposes, costing techniques which are used to control and make informed decisions to bear on cost units and cost centers in an organization. There are other topics that could be applied to cost determination and pricing of products. A study of the principles provides an in-depth knowledge and skill required for determining, analyzing and reporting cost information in small-scale businesses and other organizations. It realizes and maintains the tenets of morals and values associated with the business program as a whole.

Course Outcomes (CO):

1. Use the concepts of cost in costing and pricing of products and activities.
2. Appreciate the need to apply cost accounting principles in everyday life.
3. Adopt moral ethics and values in costing and pricing products in all spheres of life.
4. Acquire skills and abilities to aid planning, analysis, production, marketing, finance, etc. in various areas in business and personal life.
5. Use costing as a means of control and decision making in business and other spheres of life.
6. Encourage students to take higher tasks and assignments in cost accounting in the tertiary institutions particularly during the first year of study of the subject.

Course Contents:

Module-1: Meaning and Scope of Cost Accounting:

Basic Cost Concepts – Elements of Costs, Classification of Costs, Total Cost Build up and Cost Sheet, Emerging Terms viz. Life Cycle Costing, Activity Based Costing, Back Flush Costing.

Materials Control:

Meaning – Steps Involved – Materials and Inventory – Techniques of Material/Inventory Control – Valuation of Inventory – Material Losses.

Module-2:Labour Cost Control: Direct and Indirect Labour, Steps Involved – Treatment of Idle Time, Holiday Pay, Overtime etc. in Cost Accounts, Casual Workers & Out Workers, Labour Turnover, Methods of Wage Payment. IncentivePlans.

Module-3:Overheads: Meaning and Classification of Overheads – Treatment of Specific Items of Overheads in Cost Accounts – Stages Involved in Distribution of Overheads – Methods of Absorption of Overheads – Treatment of Under and Over Absorption of Overheads.

Module-4: Methods of Costing: Single Output Costing, Job Costing, Contract & Batch Costing.

Module-5: Process Costing (including Joint Products and By-products and Inter-process Profits), Operating/Service Costing. (Transport & Power House only); Reconciliation of Cost and Financial Accounts.

Text Books

1. Maheshwari, S. N. and Mittal, S. N. (2009), Cost Accounting – Theory and Problems, 22nd Revised Edition, Shri MahavirBookDepot.
2. Rajasekaran, (2010), Cost Accounting, 1st edition, PearsonEducation.

Reference Books:

1. Dutta, Mahesh, (2006), Cost Accounting - Principle Practices, 1st edition, PearsonEducation.
2. Arora, M.N., (2009), Cost Accounting, VikasPublishingHouse.
3. Lal, Jawahar and Srivastava, Seema, (2008), Cost Accounting, 4th Edition, McGraw Hill Education.
4. Jain and Narang, (2008), Cost Accounting, KalyaniPublishers.

TITLE OF COURSE: BUSINESS ACCOUNTING

COURSE CODE: BCA305B

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Business Accounting.

Introduction:

The objective of the course is to understand the basic concepts of Accounting and relate them to real life documents and business events. The course also aims to equip the students with a

strong foundation of the essential rudiments of financial accounting and help them develop a managerial perspective towards the same.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues in Business Accounting.

At the end of the course the students are able to:

1. Process business transactions based on documents commonly used by businesses
2. Identify the documents related to business transactions.

Course Contents:

Module 1:

Financial Accounting: An Overview Accounting Postulates, concepts and principles Accounting, Concepts – Conventions – Importance and scope, Accounting Principles- Double entry system – Brief overview of accounting cycle, Introduction to Balance Sheet and Income Statement, Methods – Accounting – Importance.

Module 2:

Mechanics–Journals Accounting Mechanics– Ledger Post ingand Trial Balance , Introduction - Definitions -Sale of Goods -Rendering of Services -The Use by Others of Enterprise Resources Yielding Interest-Royalties and Dividends.

Module 3:

Preparation of Financial Statements: Profit & Loss A/C.

Module 4:

Preparation of Financial Statements: Balance Sheet Basic cost concepts, Cost Determination Process Costing for materials, labor and over heads.

Text Books

1. Modern Accountancy, Vol II, Mukherjee, TMH

Reference:

1. Jain, S P. Narang, K L. (2012), Cost Accounting: Principles and Practice, 23rd. Edition, Kalyani Publishers, Ludhiana.

TITLE OF COURSE: ESP & SDP - III

COURSE CODE: GSC301

L-T-P: 0-1-0

CREDITS: 1

SYLLABUS

Introduction:

The course is on Data Interpretation, Quant Review, reasoning, English etc.

Course contains:

Module-1

- 1) Simple & Compound Interest- Basic concept of SI & CI, different formulas & their applications, concept of Growth & Contraction of Business.
- 2) Data Interpretation- Tables, pie chart, histogram, Bar chart, solution tricks & techniques.
- 3) Quant Review- Miscellaneous problems from different chapters & short cuts.
- 4) Indices & Surds- Basic concept, Formulae & their applications, Finding out the square roots, Elimination of Surds, Equation solve.
- 5) Quadratic Equation- Polynomials, degree, powers, Equation & factors Solution. Progression- Concept of AP, GP & HP

Module-2

- 1) Syllogism
- a) Logical Venn diagram b) The If Else Statement
- 2) Puzzles
- a) Seating Arrangement b) Classification
- c) Seating Arrangement with Blood relations
- 3) Machine Input-Output
- a) Pattern Based I/O
- 4) Inequality
- a) Coded Inequality, b) Jumbled Inequality, c) Conditional inequality

Module-3

- 1) Sentence Corrections
- 2) Fill the blanks with appropriate words/articles/preposition/verbs/adverbs/conjunction.
- 3) Reading Comprehension (Advance Level)
- 4) Vocabulary

Module- 4

Data interpretation Advanced Level.

Newspaper reading: The Hindu & Economic Times.

Fourth Semester Syllabus

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	CC	BCA401	Database Management System	3	1	3	5.5
2	CC	BCA402	Object Oriented Programming using Java	3	1	3	5.5
3	CC	BCA403	Formal Languages & Automata Theory	3	0	0	3
4	CC	BCA404	Discrete Mathematics	3	0	0	3
5	DE*		Elective-I	3	0	2	4
6	DE**		Elective-II	3	0	2	4
7	PTI	INT401	Internship Industrial Training	0	0	6	3
8	GS	GSC401	ESP & SDP-IV	0	1	0	1
Total							29

Students have to take one mandatory NPTEL course-based on the respective year's offered courses.

Suggestive Choice Based Subjects

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	DE*	BCA404A	Computer Graphics & Multimedia Technology	3	0	3	4.5
2	DE*	BCA404B	Foundation to Computer System Design	3	0	3	4.5
3	DE**	BCA405A	Web Development with ASP.NET	3	0	3	4.5
4	DE**	BCA405B	Statistical Modeling using R	3	0	3	4.5

TITLE OF COURSE: DATABASE MANAGEMENT SYSTEM

COURSE CODE: BCA401

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in Database, Database management system, Structured Query Language (SQL), Relational algebra and Calculus.

Introduction:

Database Management Systems (DBMS) consists of a set of interrelated data and a set of programs to access that data. They underpin any computer system and are therefore fundamental to any program to study in computer science. An understanding of DBMS is crucial in order to appreciate the limitations of data storage and application behavior and to identify why performance problems arise. Students who complete this course are expected to develop the ability to design, implement and manipulate databases. Students will apply and build databases for various day to day real life scenarios and real life applications. The course will by and large be structured but will introduce open-ended data base problems.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of Database Management System and its maintenance. To reach this goal, the following objectives need to be met:

1. Ability to build normalized databases.
2. Knowledge of Entity Relationship Modeling.
3. Familiarity with SQL, embedded SQL and PLSQL.
4. Familiarity with query processing and query optimization techniques.
5. Understanding of transaction processing.
6. Ability to handle recovery and concurrency issues.
7. Familiarity with ODBC, JDBC

Course Contents:

Module 1:

Introduction to DBMS- Concept & overview of DBMS, Data Models & database Language, Database Administrator, Database Users, architecture of DBMS, Three levels of abstraction.

Module 2:

Entity Relationship Model – Basic concepts, Design Issues, Entity-Relationship Diagram, Weak

Entity Sets, Extended E-R features.

Relational Model-

Structure of relational Databases, Relational Algebra, Relational Algebra Operations, Views, Modifications of the Database.

Module 3:

SQL and Integrity Constraints: Concept of DDL, DML, DCL, Basic Structure, Set operations, Aggregate Functions, Null Values, Domain Constraints, Referential Integrity Constraints, assertions, views, nested Sub queries.

Module 4:

Relational Database Design: -Functional Dependency, Different anomalies in designing a Database, Normalization using functional dependencies, Decomposition, Boyce-Codd Normal Form, 3NF, Normalization using multi-valued dependencies, 4NF, 5NF.

Module 5:

Transaction: -Transaction concept, transaction model, serializability, transaction isolation level, Transaction atomicity and durability, transaction isolation and atomicity.

Concurrency control and recovery system: Lock based protocol, dead lock handling, time stamp based and validation based protocol, failure. Classification, storage, recovery algorithm, recovery and atomicity, backup.

Module 6:

Internals of RDBMS:-Physical data structures, Query optimization: join algorithm, Statistics and cost based optimization.

Module 7:

File Organization & Index Structures:-File & Record Concept, Placing file records on Disk, Fixed and Variable sized Records, Types of Single-Level Index (primary secondary, clustering), Multilevel Indexes, Dynamic Multilevel Indexes using B tree and B+ tree.

Text Books

1. Silberschatz, Korth and Sudarshan, “Database System Concepts”, 6th Edition, McGraw Hill, 2010
2. Elmasri and Navathe, “Fundamentals of Database Systems”, 6th Edition, Pearson, Addison-Wesley, 2010

References

1. C.J. Date, “An Introduction to Database Systems”, 8th Edition, Addison-Wesley, 2003
2. Ramakrishnan & Gherke, Database Management Systems, 2nd Edn., McGraw
3. Connolly and Begg, “Database Systems”, 4th Edn., Addison-Wesley, 2005
4. Toby, Lightstone and Jagadish, “Database Modeling and Design”, 5th Edn, Elsevier, 2011
5. Coronel and Rob, “Database Systems”, 9th Edn., Cengage, 2011
6. IEEE / ACM Transactions on Database Systems (TODS).
7. DBMS related Journals.

TITLE OF COURSE: DATABASE MANAGEMENT SYSTEM LAB

COURSE CODE: BCA491

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts of DBMS.

Objectives:

At the end of the semester, the students should have clearly understood and implemented the following:

1. Stating a database design problem.
2. Preparing ER diagram
3. Finding the data fields to be used in the database.
4. Selecting fields for keys.
5. Normalizing the database including analysis of functional dependencies.
6. Installing and configuring the database server and the front end tools.
7. Designing database and writing applications for manipulation of data for a standalone and shared database including concepts like concurrency control, transaction roll back, logging, report generation etc.
8. Get acquainted with SQL. In order to achieve the above objectives, it is expected that each student will choose one problem. The implementation shall begin with the statement of the objectives to be achieved, preparing ER diagram, designing of database, normalization and finally manipulation of the database including generation of reports, views etc. The problem may first be implemented for a standalone system to be used by a single user. All the above

steps may then be followed for development of a database application to be used by multiple users in a client server environment with access control. The application shall NOT use web techniques. One exercise may be assigned on creation of table, manipulation of data and report generation using SQL.

Course Outcomes (CO):

1. Ability to build normalized databases.
2. Knowledge of Entity Relationship Modelling. • Familiarity with SQL, embedded SQL
3. Familiarity with query processing and query optimization techniques.
4. Understanding of transaction processing.
5. Ability to handle recovery and concurrency issues.
6. Familiarity with ODBC, JDBC.

Course Contents:

Exercises that must be done in this course are listed below:

Exercise No.1:

ER Model: An entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system.

Exercise No. 2:

EER Model: In computer science, the enhanced entity-relationship (EER) model is a high-level or conceptual data model incorporating extensions to the original entity-relationship (ER) model, used in the design of databases. It was developed by a need to reflect more precisely properties and constraints that are found in more complex databases.

Exercise No. 3:

Relational Model: The relational model for database management is a database model based on first-order 4predicate logic, first formulated and proposed in 1969 by E.F. Codd. The model uses the concept of a mathematical relation, which looks somewhat like a table of values -as its basic building block, and has its theoretical basis in set theory and first-order predicate logic.

Exercise No. 4:

1 NF: First normal form (1NF or Minimal Form) is a normal form used in database normalization. A relational database table that adheres to 1NF is one that meets a certain minimum set of criteria. These criteria are basically concerned with ensuring that the table is a faithful representation of a relation and that it is free of repeating groups.

Exercise No. 5:

2 NF: Second normal form (2NF) is a normal form used in database normalization. 2NF was originally defined by E.F. Codd in 1971. A table that is in first normal form(1NF) must

Exercise No. 6:

3 NF: The Third normal form (3NF) is an important form of database normalization. 3NF is said to hold if and only if both of the following conditions hold: • The relation R (table) is in second

normal form (2NF) • Every non-prime attribute of R is non-transitively dependent (i.e. directly dependent) on every candidate key of R.

Exercise No. 7:

BCNF: A relation R is in Boyce-Codd normal form (BCNF) if and only if every determinant is a candidate key. 4The definition of BCNF addresses certain (rather unlikely) situations which 3NF does not handle.

Exercise No. 8:

SQL-1: In this lab., we discuss basic SQL operations like creating a table, deleting a table, changing the schema of the table, primary key and foreign key constraints on a table and creating indexes on tables.

Exercise No. 9:

SQL-2: Its scope includes efficient data insert, query, update and delete, schema creation and modification, and data access control. In this lab., we discuss SQL operations for populating the tables like inserting into a table, deleting values from a table, and updating the content of the tables.

Text Books

1. Silberschatz, Korth and Sudarshan, "Database System Concepts", 6thEdition,McGraw Hill, 2010
2. Elmasri and Navathe, "Fundamentals of Database Systems", 6thEdition, Pearson, Addison-Wesley,2010
3. Ivan Bayross, "The programming language of oracle", 5thEdition, BPB Publication 2016

References

1. "Database Systems: A Practical Approach to design, Implementation and Management". Thomas Connolly, Carolyn Begg; Third Edition, Pearson Education.
2. "Fundamentals of Database Systems" Elmasri, Navathe, Pearson Education.
3. Bipin C Desai, "An Introduction to Database Systems?", Galgotia. Publications Pvt Limited, 2001
4. "An Introduction to Database Systems", C.J.Date, Pearson Education.
5. "A first course in Database Systems", Jeffrey D. Ullman, Jennifer Windon, Pearson, Education.
6. "Data Management: databases and organization", Richard T. Watson, Wiley.
7. "DataModeling Essentials", Graeme C. Simxion, Dreamtech.
8. Introduction to Data Base Management, Naveen Prakash, Tata McGraw Hill
9. "Oracle 10g manuals".

TITLE OF COURSE: OBJECT ORIENTED PROGRAMMING USING JAVA**COURSE CODE: BCA402****L-T-P: 3-1-3****CREDITS: 5.5**

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in any programming concept.

Introduction:

This course contains object oriented programming and able to explain the difference between object oriented programming and procedural programming. Be able to program using java features such as composition of data types, classes, objects, inheritance and polymorphism, file I/O, exception handling, methods, string and array manipulation, Thread, and Java applets.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of Object Oriented Programming. To reach this goal, the following objectives need to be met:

1. To know different properties of object oriented programming language, advantages and disadvantages of java over C and C++, To know different steps of java source code compilation and execution
2. To know encapsulation, polymorphism, why java is called platform independent programming language.
3. To know different access specifiers of java, finalize and garbage collection of java.
4. To know the significant of static keyword, String vs StringBuffer class and different methods of String.
5. To know different inheritance with examples.
6. To know uses of this, this(), super, super(), abstract class and interface.
7. To know member access for packages.
8. To know different ways of exception handling.
9. To know different ways of implementing concept of multithreading, problems in multithreading and their solutions.
10. To know benefits and uses of applet and implementation of applet.
11. To know different components of swing, event handling and layouts in swing.

Course Contents:

Module-1: Introduction

Different Programming Paradigms, Why use the Object Oriented Paradigm

Object Oriented Principles, Concept of Class, Object, inheritance, encapsulation, and polymorphism

Module-2: An Overview of Java:

Advantage of JAVA , Feature of JAVA in object oriented programming Concept of byte-code & JVM, Basic JAVA Syntax, Basic Data Type, Control Statement using Java, Concept of Arrays in JAVA, Concept of Method in JAVA

Module-3: Introducing Class & Method:

Objects and classes in java, defining classes

Module-4: Building our own Classes:

Analyzing the class, Methods to access the data, Method Overloading, Static Method, Constructors, Object Construction, Default Constructors, Constructor Overloading, and Garbage Collection

Module-5: Inheritance & other features:

Inheritance, Abstract Classes, Final Classes in JAVA, Access specifiers in java, Method overriding, Interface, Polymorphism in java

Module-6: Packages:

Concept of Package in JAVA, Member access for packages

Module-7: Exception Handling:

Concept of exception, Exception, Hierarchy, Catching Exceptions, Declaring you own Exception

Module-8: String & Thread:

Basic string handling concepts, Discuss Basic method of String, Life Cycle of a Thread, Create Thread, Thread Methods, Multi Thread

Module-9: Applet & Swing:

Concept of Applet, Life Cycle of an Applet, Basic Applet Programming, Concept of Swing , Swing Controls, Event handling And Listner in Swing Swing Layout Manager

Text Books

1. Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007.
2. Michael Blaha and James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education, 2005.

3. PatricNaughton , Herbert Schildt, "Java 2 Complete Reference", Tata McGraw Hill, 1999.

References

1. Joshua Bloch, “Effective Java”, Addison-Wesley; 3rd Edition, 2018.

2. Bruce Eckel, “Thinking in Java”, Prentice Hall; 4th Edition, 2006.

3. Erich Gamma, Richard Helm, Ralph Johnson & John Vlissides, “Design Patterns: Elements of Reusable Object-oriented Software”, Pearson Education India, 2004.

TITLE OF COURSE: OBJECT ORIENTED PROGRAMMING USING JAVA LAB

COURSE CODE: BCA492

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students must have already registered for the course, basic computation.

Objectives: The model of object oriented programming: abstract data types, encapsulation, inheritance and polymorphism Fundamental features of an object oriented language like Java: object classes and interfaces, exceptions and libraries of object collections. How to take the statement of a business problem and from this determine suitable logic for solving the problem; then be able to proceed to code that logic as a program written in Java. How to test, document and prepare a professional looking package for each business project using javadoc.

Course Outcomes: Students will also be able to handle software development models through rational method. After completion of course student will be able to generate test cases for software testing.

Course Contents:

Unit 1: Concepts of object oriented programming language, Major and minor elements, Object, Class, relationships among objects, aggregation, links, relationships among classes-association, aggregation, using, instantiation, meta-class, grouping constructs.

Unit 2: Difference between OOP and other conventional programming – advantages and disadvantages. Class, object, message passing, inheritance, encapsulation, polymorphism.

Unit 3: Basic concepts of java programming – advantages of java, byte-code & JVM, data types, access specifiers, operators, control statements & loops, array, creation of class, object, constructor, finalize and garbage collection, use of method overloading, this keyword, use of objects as parameter & methods returning objects, call by value & call by reference, static

variables & methods, garbage collection, nested & inner classes, basic string handling concepts-String (discuss charAt() , compareTo(), equals(), equalsIgnoreCase(), indexOf(), length() , substring(), toCharArray() , toLowerCase(), toString(), toUpperCase() , trim() , valueOf() methods) & StringBuffer classes (discuss append(), capacity(), charAt(), delete(), deleteCharAt(), ensureCapacity(), getChars(), indexOf(), insert(), length(), setCharAt(), setLength(), substring(), toString() methods), concept of mutable and immutable string, command line arguments, basics of I/O operations – keyboard input using BufferedReader& Scanner classes.

Unit 4: Super class & subclasses including multilevel hierarchy, process of constructor calling in inheritance, use of super and final keywords with super() method, dynamic method dispatch, use of abstract classes & methods, interfaces. Creation of packages, importing packages, member access for packages.

Unit 5: Exception handling basics, different types of exception classes, use of try & catch with throw,

throws & finally, creation of user defined exception classes. Basics of multithreading, main thread, thread life cycle, creation of multiple threads, thread priorities, thread synchronization, inter-thread

communication, deadlocks for threads, suspending & resuming threads.

Unit 6: Basics of applet programming, applet life cycle, difference between application & applet programming, parameter passing in applets, concept of delegation event model and listener, I/O in

applets, use of repaint(), getDocumentBase(), getCodeBase() methods, layout manager (basic concept), creation of buttons (JButton class only) & text fields.

List of Experiments

Experiment 1: Write a program that reads the radius of a circle as input taken dynamically while running and then computes the circumference and area.

Experiment 2: Write a java program to print the following pattern using “for” loops

```
1
2 3
4 5 6
```

Experiment 3: Write a java program to eliminate the duplicate elements from the given set of input elements

Sample input: 1 2 3 4 4 5 3 2

Sample Output: 1 2 3 4 5

Experiment 4: Write a java program to find the nth smallest number in the given set of elements

Sample input 2 4 3 5 8 7 N=3

Sample output: The third smallest element is:4

Experiment 5: Write a java program to sort the elements of the matrix in row wise

Sample input 1 3 2 sample output: 1 2 3

6 5 4 4 5 6

9 7 8 7 8 9

Experiment 6: Write a java program to find the N prime numbers

Sample input:7

Sample output: 2 3 5 7 11 13 17

Experiment 7: Write a program to use concept of packages with access specifiers.

Experiment 8. Write a code to implement user defined exception.

Experiment 9: Write a program to implement thread in java.

Experiment 10: How can you change priority of a thread?

Experiment 11: When synchronization problem is occurred in thread program? Explain with code.

Experiment 12: Explain the life cycle of Applet with program.

Experiment 13: Write down the importance of Applet program.

Experiment 14: Design a swing application to store student data in a file. Use JFrame, JPanel, JLabel, JTextField, JButton to design the application.

Experiment 15: Write down the use of use of repaint(), getDocumentBase(), getCodeBase() methods.

Experiment 16: Explain use of different events in java.

Experiment 17: Explain use of different Layout manager.

References

1. Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson Education, 2007.
2. Michael Blaha and James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education, 2005.
3. Patric Naughton , Herbert Schildt, "Java 2 Complete Reference", Tata McGraw Hill, 1999.
4. Joshua Bloch, "Effective Java", Addison-Wesley; 3rd Edition, 2018.
5. Bruce Eckel, "Thinking in Java", Prentice Hall; 4th Edition, 2006.
6. Erich Gamma, Richard Helm, Ralph Johnson & John Vlissides, "Design Patterns: Elements of Reusable Object-oriented Software", Pearson Education India, 2004.

TITLE OF COURSE: FORMAL LANGUAGES & AUTOMATA THEORY**COURSE CODE: BCA403****L-T-P: 3-0-0****CREDITS: 3**

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in mathematics like set theory.

Introduction:

This course examines formal language and automata theory concepts. The Topics to be covered (tentatively) include: Finite state machine, Regular language and finite automata, Context free language and pushdown automata and Turing Machine.

Course Outcomes (CO):

This course introduces the formal language & automata theory. Starting with the most primitive computing device, a finite automaton, the course gradually introduces additional components to the device to enhance its computing power. The course also introduces students to the twin concepts of languages and grammars that correspond to classes of computing devices. Finally the course introduces the idea of a universal computing device and brings out theoretical limits of the very idea of computing. To reach this goal, the following objectives need to be met:

1. To design deterministic and nondeterministic finite automata like Mealy machine, Moore machine, DFA and NFA.
2. To construct regular expressions and grammars.
3. To convert context free grammars to normal forms and eliminate ambiguities.
4. To design context free grammar recognizable language like deterministic and non-deterministic PDA.
5. To design deterministic and nondeterministic automata and Turing machines. To recognize unsolvable problems and limitations of computing. Get familiarity with the seminal works of Turing and Chomsky.

Course Contents:

Module-1: Finite Automata: Definition, Application of finite automata. Concept and design of Moore & Mealy machine, Mealy to Moore conversion and vice versa, minimization. Deterministic finite automaton and non-deterministic finite automaton. Transition diagrams and Language recognizers. NFA with ϵ transitions - Significance, acceptance of languages. Conversions and Equivalence: Equivalence between NFA with and without ϵ transitions. NFA to DFA conversion.

Module-2: Regular Language and Grammar: Regular sets. Regular expressions, identity rules. Arden's theorem state and prove. Constructing finite Automata for a given regular expressions, Regular string accepted by NFA/DFA, Pumping lemma of regular sets. Closure properties of

regular sets (proofs not required). Grammar Formalism: Regular grammars-right linear and left linear grammars. Equivalence between regular linear grammar and FA.

Module-3: Context Free Language and Grammar: Definition of Context Free Grammars, Derivation trees, sentential forms. Right most and leftmost derivation of strings. Ambiguity in context free grammars. Minimization of Context Free Grammars. Chomsky normal form and Greibach normal form. Pumping Lemma for Context Free Languages. Enumeration of properties of CFL (proofs omitted). Closure property of CFL, Ogden's lemma & its applications.

Module-4: Push Down Automata: Push down automata of definition. Acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, inter-conversion. (Proofs not required). Introduction to DCFL and DPDA.

Module-5: Turing Machine: Definition, construction, types of TM. Concepts of Universal Turing machine and Halting problem.

Text Books

1. Mishra and Chandrashekar, "Theory of Computer Science , Automata Languages and computation", 2nd edition, PHI.
2. Peter Linz, "Introduction to Formal Language and Automata", 5th edition, Jones and Bartlett's Publications.

References

1. C.K.Nagpal, "Formal Languages and Automata Theory", Oxford.
2. Hopcroft H.E. and Ullman J. D , "Introduction to Automata Theory Language and Computation", Pearson Education.
3. John C Martin, "Introduction to languages and the Theory of Computation", TMH

TITLE OF COURSE: DISCRETE MATHEMATICS

COURSE CODE: BCA404

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Introduction:

To introduce students to language and methods of the area of Mathematics of computing. The focus of the module is on basic mathematical concepts in discrete mathematics and on applications of discrete mathematics in algorithms and data structures. To show students how

mathematics can be used in modern computer science (with the focus on algorithmic applications).

Course Outcomes (CO):

1. Understand the notion of mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving.
2. Understand the basics of discrete probability and number theory, and be able to apply the methods from these subjects in problem solving.
3. Be able to use effectively algebraic techniques to analyse basic discrete structures and algorithms.
4. Understand asymptotic notation, its significance, and be able to use it to analyse asymptotic performance for some basic algorithmic examples.
5. Understand some basic properties of graphs and related discrete structures, and be able to relate these to practical examples.
6. Introduction to combinatorics: counting techniques, pigeonhole principle, inclusion-exclusion.
7. Recurrence relations, solving recurrences using generating functions.
8. Master Theorem for solving recurrences.
9. Graphs. Basic graph algorithms. Trees. Applications of graphs.
10. Applications of linear algebra and matrix algebra in algorithms (e.g., in web searching).
11. Algorithmic applications of random processes and Markov chains, for example, cover time in graphs and card shuffling.
12. Partitions, enumerations with symmetries.

Course Contents:

Module 1 (8L)

Introduction to Propositional Calculus: Propositions, Logical Connectives, Conjunction, Disjunction, Negation and their truth table. Conditional Connectives, Implication, Converse, Contrapositive, Inverse, Bi-conditional statements with truth table, Logical Equivalence, Tautology, Normal forms-CNF, DNF; Predicates and Logical Quantifications of

propositions and related examples.

Module 2 (8L)

Theory of Numbers: Well Ordering Principle, Divisibility theory and properties of divisibility; Fundamental theorem of Arithmetic; Euclidean Algorithm for finding G.C.D and some basic properties of G.C.D with simple examples; Relation and Lattices: POSET, Hasse Diagram, Minimal , Maximal, Greatest and Least elements in a POSET.

Module 3 (10L)

Counting Techniques: Permutations, Combinations, Binomial coefficients, Pigeon- hole Principle, Principles of inclusion and exclusions; Recurrence relations: Formulation/Modelling of different counting problems in terms of recurrence relations, Solution of linear recurrence relations with constant coefficients (upto second order) by (i) The Characteristic roots method (iii) Generating functions method.

Module 4 (10)

Graph Theory Concepts Graphs, sub-graphs, cyclic graphs. Trees, spanning trees, binary trees, Algorithms- Prim's, Kruskal, Isomorphism, homomorphism

Text Books

1. Rosen: Discrete mathematics and application(7thed),McGraw Hill
2. Satyanarayana­am Prasad :discrete mathematics and graph theory, PHI
3. Discrete Structures and Graph Theory, Rao, Sci tech
4. Mathematical Foundations, Vijayarangan, Scitech
5. Discrete Structures and Graph Theory, Rathor, EPH.

References

1. Kishorshinde: Discrete Structure, Everest publishing house
2. HariParihar&Ritu Agarwal, discrete mathematical structures, ashirwad

TITLE OF COURSE: COMPUTER GRAPHICS & MULTIMEDIA TECHNOLOGY**COURSE CODE: BCA404A****L-T-P: 3-0-3****CREDITS: 4.5**

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and basic mathematics.

Introduction:

Computer Graphics course presents an introduction to computer graphics designed to give the student an overview of fundamental principles. It covers the fundamental concepts in creating graphical images on the computer. Computer graphics uses ideas from Art, Mathematics, and Computer Science to create images. Course work stresses the reduction of concepts to practice in the form of numerous programming assignments. The course will include an overview of common graphics hardware, 2D and 3D transformations and viewing, and basic raster graphics concepts such as scan-conversion and clipping. Methods for modeling objects as polygonal meshes or smooth surfaces, and as rendering such as hidden-surface removal, shading, illumination, and shadows will be investigated.

Multimedia course provides mainstreaming the technological media within what is called “Multimedia” is the pattern which led to infinite applications of computer technologies. The concept of this technology came into being with the appearance of sound cards, then compact disks, then came the use of digital camera, then the video which made computer an essential educational tool. Nowadays, multimedia expanded to become a field on its own.

Course Outcomes (CO):

After completion of the course, students will able:

1. To know and be able to understand the core concepts of computer graphics.
2. To know and be capable of using OpenGL to create interactive computer graphics.
3. To know and be able to understand a typical graphics pipeline.
4. To know and be able to make interactive graphics applications in C++ using one or more graphics application programming interfaces.
5. To know and be able to demonstrate an understanding of the use of object hierarchy in graphics applications.
6. To know and be able to write program functions to implement visibility detection.
7. To know and be able to make pictures with their computer.
8. To know and be able to describe the general software architecture of programs that use 3D computer graphics
9. To know the pictorial representation of various points in a image

Course Contents:

Module-1: Introduction to computer graphics & graphics systems

Overview of computer graphics, representing pictures, preparing, presenting & interacting with pictures for presentations; Visualization & image processing; RGB color model, direct coding, lookup table; storage tube graphics display, Raster scan display, 3D viewing devices, Plotters, printers, digitizers, Light pens etc.; Active & Passive graphics devices; Computer graphics software.

Scan conversion

Points & lines, Line drawing algorithms; DDA algorithm, Bresenham's line algorithm, Circle generation algorithm; Ellipse generating algorithm; scan line polygon, fill algorithm, boundary fill algorithm, flood fill algorithm.

Module-2: 2D transformation & viewing

Basic transformations: translation, rotation, scaling; Matrix representations & homogeneous coordinates, transformations between coordinate systems; reflection shear; Transformation of points, lines, parallel lines, intersecting lines. Viewing pipeline, Window to viewport coordinate transformation, clipping operations, point clipping, line clipping, clipping circles, polygons & ellipse.

3D transformation & viewing

3D transformations: translation, rotation, scaling & other transformations. Rotation about an arbitrary axis in space, reflection through an arbitrary plane; general parallel projection transformation; clipping, viewport clipping, 3D viewing

Module-3: Curves

Curve representation, surfaces, designs, Bezier curves, B-spline curves, end conditions for periodic B-spline curves, rational B-spline curves. Hidden surfaces, Depth comparison, Z-buffer algorithm, Back face detection, BSP tree method, the Painter's algorithm, scan-line algorithm; Hidden line elimination, wire frame methods, fractal - geometry. Color & shading models Light & color model; interpolative shading model; Texture.

Module-4: Multimedia

Introduction to Multimedia: Concepts, uses of multimedia, hypertext and hypermedia; Image, video and audio standards. Audio: digital audio, MIDI, processing sound, sampling, compression. Video: MPEG compression standards, compression through spatial and temporal redundancy, inter-frame and intra-frame compression. Animation: types, techniques, key frame animation, utility, morphing. Virtual Reality concepts.

Text Books

1. Hearn, Baker – “Computer Graphics (C version 2nd Ed.)” – Pearson education
2. Z. Xiang, R. Plastock – “Schaum's outlines Computer Graphics (2nd Ed.)” – TMH
3. D. F. Rogers, J. A. Adams – “Mathematical Elements for Computer Graphics (2nd Ed.)” – TMH
4. Mukherjee, Fundamentals of Computer graphics & Multimedia, PHI
5. Sanhker, Multimedia –A Practical Approach, Jaico

References

1. Buford J. K. – “Multimedia Systems” – Pearson Education
2. Andleigh&Thakrar, Multimedia, PHI
3. Mukherjee Arup, Introduction to Computer Graphics, Vikas
4. Hill, Computer Graphics using open GL, Pearson Education

TITLE OF COURSE: COMPUTER GRAPHICS & MULTIMEDIA TECHNOLOGY LAB

COURSE CODE: BCA494A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and basic mathematics.

Introduction:

This course presents an introduction to computer graphics designed to give the student an overview of fundamental principles. It covers the fundamental concepts in creating graphical images on the computer. Computer graphics uses ideas from Art, Mathematics, and Computer Science to create images. Course work stresses the reduction of concepts to practice in the form of numerous programming assignments. The course will include an overview of common graphics hardware, 2D and 3D transformations and viewing, and basic raster graphics concepts such as scan-conversion and clipping. Methods for modeling objects as polygonal meshes or smooth surfaces, and as rendering such as hidden-surface removal, shading, illumination, and shadows will be investigated.

Course Outcomes (CO):

After completion of the course, students will able:

1. To know and be able to understand the core concepts of computer graphics.
2. To know and be capable of using OpenGL to create interactive computer graphics.
3. To know and be able to understand a typical graphics pipeline.
4. To know and be able to make interactive graphics applications in C++ using one or more graphics application programming interfaces.
5. To know and be able to demonstrate an understanding of the use of object hierarchy in graphics applications.

6. To know and be able to write program functions to implement visibility detection.
7. To know and be able to make pictures with their computer.
8. To know and be able to describe the general software architecture of programs that use 3D computer graphics

Course Contents:**Experiments:**

1. Study of basic graphics functions defined in “graphics.h”
2. Write a program to draw a any geometrical figure.
3. Write a program to draw a line using Bresenham’s algorithm
4. Write a program to draw a line using DDA algorithm
5. Write a program to draw a line using Mid point algorithm
6. Write a program to draw a circle using Midpoint algorithm
7. Write a program to draw a Ellipse using Midpoint algorithm

Recommended Systems/Software Requirements:

1. Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
2. Turbo C or TC3 compiler in Windows XP or Linux Operating System.

Text Books

1. Hearn, Baker – “Computer Graphics (C version 2nd Ed.)” – Pearson education
2. Z. Xiang, R. Plastock – “ Schaum’s outlines Computer Graphics (2nd Ed.)” – TMH
3. D. F. Rogers, J. A. Adams – “Mathematical Elements for Computer Graphics (2nd Ed.)” – TMH
4. Sanhker, Multimedia –A Practical Approach, Jaico

References

5. Buford J. K. – “Multimedia Systems” – Pearson Education
6. Andleigh&Thakrar, Multimedia, PHI
7. Mukherjee Arup, Introduction to Computer Graphics, Vikas
8. Hill, Computer Graphics using open GL, Pearson Education

TITLE OF COURSE: FOUNDATION TO COMPUTER SYSTEM DESIGN

COURSE CODE: BCA404B

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and computer architecture.

Introduction:

The Computer Architecture and Organization, Operating Systems, and Compilers are three fundamental pillar courses for both Computer Science and Engineering; and, Electrical and Electronics engineering students. The current course presents a cross-layer view of the three pillars, which help the student appreciate the contributions, interactions and challenges related to each of these pillars from the view of the total systems engineering.

Course Outcomes (CO):

Learn how a computer works with this basic, but detailed course on the foundations of computer design. In this course you will learn how a computer really works and by the end you will be able to understand the design of a programmable central processing unit. This class is the foundation of a computer science curriculum.

1. Learn binary and hexadecimal numbers
2. Understand logic gates
3. Recognize multiplexers decoders and the ALU
4. Learn how to design memory chips and CPU registers
5. Build a basic central processing unit

Course Contents:

Module-1: Introduction: Functional components of a computer system, Interaction among functional components.

Module-2: Design of Arithmetic and Logic Unit: Digital logic gates, Boolean algebra, Simplification of Boolean Expressions, Combinational logic circuits – Encoder, Decoder, Multiplexer and Demultiplexer; Representation of integer data, Integer adders, Integer multipliers, Design of integer unit, Floating point representation of real data, Floating-point adder/subtractor, Floating-point multiplier, Design of Floating point unit.

Module-3: Design of Register File and Memory Unit: Flip-flops, Synchronous sequential circuits – Registers and Counters; Memory unit.

Module-4: Generation and Execution of Machine Code: Instruction set architecture of a simple CPU, Assembler, Code generation, Application binary interface, Microarchitecture of CPU, Execution of machine code.

Module-5: Processing of High Level Language Code: Virtual machine, Interpreter, A simple compiler, Fundamentals of operating system.

Text Books

1. N. Nisan and S. Schocken, The Elements of Computing Systems – Building a Modern Computer from First Principles, The MIT Press, 2005.
2. C.H. Roth and L.L.Kinney, Fundamentals of Logic Design, 7th Edition, Cengage Learning, 2014.
3. C.Hamacher, Z.Vranesic, S.Zaky and N.Manjikian, Computer Organization and Embedded Systems, 6th Edition, McGraw-Hill, 2012.

References

1. S. Brown and Z Vranesic, Fundamentals of Digital Logic with VHDL Design, McGraw-Hill Education, 3rd Ed, 2009.

TITLE OF COURSE: FOUNDATION TO COMPUTER SYSTEM DESIGN LAB

COURSE CODE: BCA494B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students must have already registered for the course, basic computer.

Objectives: Students will be capable to acquire the basic computer development skill through various operating system. They will also be able to ensure the quality of software through basic computer with various protocol based environment.

Course Outcomes:

1. Students will also be able to handle computer system.
2. After completion of course student will be able to implement basic HTML,MS word MS Excel, MS power point.

Course Contents:

Module-1: Use of DOS system in computer

Module-2: Implementation of MS word file in computer system

Module-3: Implementation of MS word file in computer system

Module-4: Implementation of MS power point file in computer system

Module-5: Implementation of HTML file in computer system

Module-6: Implementation of flash file in computer system

List of Experiments

1. Create new folder, Delete folder, Update folder name and create txt file and update the text file in DOS operating System.

2. Create a new folder and do the following:

Make a word document in it.

Make an Excel document in it.

Make a new folder in it

Rename the initial folder

Move the initial folder

Copy the initial folder.

Delete the initial folder

3. Implement the various well known features of Windows operating system such as Notepad, WordPad, Paint, System tools, Entertainment etc. enclosed in Start→Programs→Accessories.

4. Implement various display properties by right clicking on the Windows Desktop.

5. Explore the taskbar of Windows.

6. Set the wall paper and screen saver.

7. Set the data/time.

MS-WORD

8. Create a document and

Put Bullets and Numbers

Apply various Font parameters.

Apply Left, Right, and Centre alignments.

Apply hyperlinks

Insert pictures

Insert ClipArt

Show the use of WordArt

Add Borders and Shading

Show the use of Find and Replace.

Apply header/footers

9. CREATE ANY DOCUMENT AND SHOW THE USE OF FILE→VERSIONS.

10. CREATE ANY DOCUMENT AND SHOW THE DIFFERENCE BETWEEN PASTE AND PASTE SPECIAL.

11. Create a document to show the use of Washout/Watermark.

12. Implement the concept of mail merge.

13. Implement the concept of macros.

14. Implement the concept of importing a file/document.

15. Implement the concept of merging the documents.

16. Create a student table and do the following:

Insert new row and fill data

Delete any existing row

Resize rows and columns

Apply border and shading

Apply merging/splitting of cells

Apply sort

Apply various arithmetic and logical formulas.

PRESENTATION SOFTWARE- MS-POWERPOINT

17. Make a presentation of College Education System using

Blank Presentation

From Design Template

From Auto Content Wizard

18. Make a presentation on “Wild Life” and apply the following:

Add audio and video effects

Apply various Color Schemes

Apply various animation schemes.

Apply Slide Show

19. Create any webpage using following HTML tags:

Background Colour

Font (Colour, Size, Face)

Bold/Italic/Underline

Big/Small

H1, H2, etc.

Marquee

Ordered/Unordered List

Data lists

20. Create Employee Table and apply various operations on it using HTML. Also put Border around the table.

21. Create Internal and External Hyperlinks in a Webpage.

22. Implement the concept of Frames in a Webpage.

23. Insert an image in a Webpage.

24. Design Home page of your Institute

25. Design Web page for tourism spots in your area

26. Use animation of image in a web page

27. Insert a table and perform table handling in web page

FrontPage

Use front page to create a new page using different text styles

Use front page to create a new page showing text and table box

Use front page to create a new page using clip art gallery

Use front page to create a new page using image from a scanner

Use front page to create a web page of your institute

Use front page to create a web page showing tourism spots of your area

Use front page to use table handling in web page

1. Create a blank flash document
2. create a flash movie and customize the movie setting
3. Set frame rate, background color, dimensions to define size of movie.
4. Use of zoom tool from tool box.
5. Creating a shape with oval tool.
6. Add text to a button.
7. Make button symbol interactive (use of edit symbol button).
8. Add key frames to a button symbol time line.
9. Add sound to a button.
10. Verify changes with test movie.
11. Add layers to a movie.
12. Animate text with tweening.
13. Add a text field containing a string of text that will change every time a user presses button.
14. Add action script to the button enabling script to change the contents of the text field every time a user clicks the button.
15. Test the movie with flash player, return to main window and save the file.
16. Embed a flash movie in a web page.

Text Books

1. N. Nisan and S. Schocken, The Elements of Computing Systems – Building a Modern Computer from First Principles, The MIT Press, 2005.
2. C.H. Roth and L.L.Kinney, Fundamentals of Logic Design, 7th Edition, Cengage Learning, 2014.
3. C.Hamacher, Z.Vranesic, S.Zaky and N.Manjikian, Computer Organization and Embedded Systems, 6th Edition, McGraw-Hill, 2012.

References

1. S. Brown and Z Vranesic, Fundamentals of Digital Logic with VHDL Design, McGraw-Hill Education, 3rd Ed, 2009.

TITLE OF COURSE: WEB DEVELOPMENT WITH ASP.NET

COURSE CODE: BCA405A

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in HTML, CSS, and Other programming knowledge.

Introduction:

To learn and understand Web design. It is a process of conceptualizing, planning, and building a collection of electronic files that determine the layout, color, text styles, structure, graphics, images, and use of interactive features that deliver pages to your site.

Course Outcomes (CO):

1. Learn the technologies of the .NET framework
2. Know the object oriented aspects of C#
3. Be aware of application development in .NET
4. Learn web based applications on .NET(ASP.NET)

Course Contents:

Module-1: Concepts of networking, Web and HTML. Introduction with Web, Network, Website, Server, Client side, Server side and other terms related to basic website designing concept. Introduction with HTML language.

Module-2: Introduction with .net. Introduction of Microsoft .net, Explain features and phases of the objectoriented approach. (C#) Basic Syntax, Reading and writing to a console, Data Types, Type Conversion, Variables, Constants.

Module-3: SQL Server, Introduction with SQL Server, Role of a Database Server, SQL language, Working With Database (Table concepts), SQL query (Data Definition Language, Data Manipulation Language, Data Control Language)

Module-4: Database Integration in ASP.NET Connectivity between web pages and data base with the help of Internal and external data source.

Module-5: Hands on C# language. Introduction of C# and programing basic of C#, Programs on different problems in C#. Introduction with core PHP, Core PHP introduction and programming concepts. Quick start with PHP programs. Database Integration in PHP.

Text Books

1. Learning Web Design, Book by Jennifer Niederst Robbins
2. NET 4.5 Programming Black Book, Kogent learning solutions inc.
3. Professional PHP 6, Publisher: Wiley

References

1. Desktop PC with minimum of 166 MHZ or faster processor with at least 1 GB RAM and 160 GB disk space.
2. Visual studio 2012, Microsoft sql server 2008 R2

TITLE OF COURSE: WEB DEVELOPMENT WITH ASP.NET LAB

COURSE CODE: BCA495A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Objectives:

To learn and understand Web design. It is a process of conceptualizing, planning, and building a collection of electronic files that determine the layout, color, text styles, structure, graphics, images, and use of interactive features that deliver pages to your site.

Course Outcomes (CO):

1. Apply critical thinking and problem solving skills required to successfully design and implement a web site.
2. Demonstrate the ability to analyze, identify and define the technology required to build and implement a web site.
3. Apply critical thinking and problem solving skills required to successfully design and implement a web site.
4. Demonstrate the ability to analyze, identify and define the technology required to build and implement a web site.
5. Demonstrate knowledge of artistic and design components that are used in the creation of a web site.
6. Utilize and apply the technical, ethical and interpersonal skills needed to function in a cooperative environment.

Course Contents:

1. Concepts of networking, Web and HTML. Introduction with Web, Network, Website, Server, Client side, Server side and other terms related to basic website designing concept. Introduction with HTML language.
2. Introduction with .net. Introduction of Microsoft .net, Explain features and phases of the object-oriented approach. (C#) Basic Syntax, Reading and writing to a console, Data Types, Type Conversion, Variables, Constants.
3. SQL Server, Introduction with SQL Server, Role of a Database Server, SQL language, Working With Database (Table concepts), SQL query (Data Definition Language, Data Manipulation Language, Data Control Language)
4. Database Integration in ASP.NET Connectivity between web pages and data base with the help of Internal and external data source.

Exercises that must be done in this course are listed below:

Exercise No.1: Create a form in HTML for entering value for some specific fields. (Registration Page)

Exercise No.2: Create table in SQL for storing data of registration page. (Using sql query)

Exercise No.3: Create a webpage to show the data which is entered in sql tables through registration page.

Exercise No. 4: Create a web page to file upload option, so user can upload document on website.

Exercise No. 5: Create a webpage to show the uploaded document.

Exercise No. 6: assemble all the web page to create a website for a specific organization. (Minor project).

Exercise No. 7: Create master page for previous developed pages.

Exercise No. 8: Apply validators for all fields which are used in previous developed pages.

Exercise No. 9: Major project.

Text Book:

1. Learning Web Design, Book by Jennifer Niederst Robbins
2. NET 4.5 Programming Black Book, Kogent learning solutions inc.

Recommended Systems/Software Requirements:

1. Desktop PC with minimum of 166 MHZ or faster processor with at least 1 GB RAM and 160 GB disk space.
2. Visual studio 2012, Microsoft sql server 2008 R2

TITLE OF COURSE: STATISTICAL MODELING USING R

COURSE CODE: BCA405B

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Basic knowledge of statistics and Proficiency with algorithms. Along with that some knowledge of data base handling is also required. Critical thinking and problem solving skills are important.

Introduction: This course gives students an opportunity to use the public domain and free software R to perform statistical computing. The R language provides a rich environment for working with data, especially for statistical modeling and graphics. The course emphasis is on data manipulation and basic statistical analysis. We will cover exploratory data analysis, programming, classical testing, categorical data analysis, and regression. Students will identify appropriate statistical methods for the data or problems and conduct their own analysis using the R environment. This is a hands-on, project-based course to enable students to develop skills and to solve statistical problems using R.

Course Outcomes (CO):

The goal of this course is to provide a supportive, hands-on environment for students to use R to conduct basic statistical analysis.

At the end of this course students will be able to:

1. Create and modify R datasets
2. Write their own R functions and use available packages in R
3. Create figures and plots using R

4. Perform and interpret parametric one- and two-sample tests using R
5. Perform and interpret nonparametric one- and two-sample tests using R
6. Perform and interpret chi-square analysis using R
7. Perform and interpret multiple linear regressions using R
8. Perform and interpret one-way ANOVA using R
9. Perform and interpret logistic and survival analyses using R

Course Contents:

Module 1 - Getting Started and Basics: An introductory R session, R as a calculator, Vectors and matrices, Getting help and loading packages, Data entry and exporting data.

Module 2 - Exploratory Data Analysis with R: Summary statistics, Graphics in R, Probability and Distribution.

Module 3 - R Programming: R as a programming language, grouping, loops and conditional execution, Functions.

Module 4 - Classical tests: One-sample tests, Two-sample tests, Tests on more than two samples.

Module 5 - Regression and correlation: Simple regression and correlation, multiple regression, Regression diagnostics.

Module 6 - Tabular data and analysis of Categorical data:

Single proportion, two independent proportions, K proportions.

Module 7 - Logistic Regression and Survival Analysis: Perform logistic regression in R Produce Kaplan-Meier curves in R, Survival analysis in R

Text Books

1. Software for Data Analysis: Programming with R (Statistics and Computing), John M. Chambers, Springer.
2. R programming for Data Science, Roger D. Peng.

References

1. An Introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics. W. N. Venables, D.M. Smith and the R Development Core Team. Version 3.0.1 (2013-05-16).

TITLE OF COURSE: STATISTICAL MODELING USING R LAB

COURSE CODE: BCA495B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Basic knowledge of statistics and Proficiency with algorithms. Along with that some knowledge of data base handling is also required. Critical thinking and problem solving skills are important.

Objectives: Optimize business decisions and create competitive advantage with Big Data analytics

- Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
- Introducing Java concepts required for developing map reduce programs
- Derive business benefit from unstructured data
- Introduce programming tools PIG & HIVE in Hadoop ecosystem.
- Developing Big Data applications for streaming data using Apache Spark

Course Outcomes (CO):

- Preparing for data summarization, query, and analysis.
- Applying data modeling techniques to large data sets
- Creating applications for Big Data analytics
- Building a complete business data analytic solution

Course Contents:

Module-1: Getting Ready to Use R and Hadoop.

Module-2: Writing HadoopMapReduce Programs

Module-3: Integrating R and Hadoop

Module-4: Using Hadoop Streaming with R

Module-5: Learning Data Analytics with R and Hadoop

.

List of Experiments

1. Implement the following Data Structures in Java
 - a) Linked Lists
 - b) Stacks
 - c) Queues
 - d) Set
 - e) Map
2. Perform setting up and Installing Hadoop in its three operating modes:
 - Standalone
 - Pseudo Distributed

- Fully Distributed
- 3. Implement the following file management tasks in Hadoop:
 - Adding files and directories
 - Retrieving files
 - Deleting Files
- 4. Run a basic Word Count Map Reduce Program to understand Map Reduce Paradigm.
- 5. Write a Map Reduce Program that mines Weather Data.
- 6. Write a Map Reduce Program that implements Matrix Multiplication..
- 7. Install and Run Pig then write Pig Latin scripts to sort, group, join, project and filter the data.
- 8. Install and Run Hive then use Hive to create, alter and drop databases, tables, views, functions and Indexes.
- 9. Write a program to analyze the Web server log stream data using Apache Flume Framework.
- 10. Write a program to implement combining and partitioning in hadoop to implement a custom partitioner and Combiner.

References

1. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly.
2. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss.

TITLE OF COURSE: ESP & SDP - IV

COURSE CODE: GSC401

L-T-P: 0-1-0

CREDITS: 1

Introduction:

The course is on GK & CA, reasoning, Permutation & Combination, Geometry, Probability English etc.

Course contains:

Module-1

GK & CA

Modern History & National Movement.

Indian Geography at a glance (Physical, Regional & Economic)

Miscellaneous: calendar etc. capitals of countries, currency of countries, important dates, Sports football, hockey etc. recent events & awards too.

Important books & authors, Important Hydropower dams, atomic power plant s, important national parks, Minster & portfolio & constituencies, Population census, Persons in news -most famous, popular recent only, Important dances & festivals of Indian states,

International Head Quarters & world organization, Important president & pm elected from various countries, Important about banks like payment banks, small banks & license system, Awards,

Sports, Books & author, National & International affairs. Current Affairs.

SKILL DEVELOPMENT FOR PROFESSIONALS IV

Module-1

- 1) Permutation & Combination.
 - 2) Probability- basic concepts of probability, different theorems & applications, binomial, poisson & normal Distributions.
 - 3) Geometry- Concept of different shapes like triangle, quadrilateral, rectangle, square, circle etc. different theorems & their applications.
 - 4) Mensuration- Formulae on triangles, square, Rhombus, parallelogram, sphere, circle, cone, pyramid etc.
- Application based problem solving. Coordinate Geometry- Locus, Straight lines, Circle etc.

Module-2

- 1) Seating Arrangement
 - a) Circular seating arrangement
 - b) Square seating Arrangement
 - c) Line Arrangement
- 2) Calendar And Clock
- 3) Miscellaneous Problems

Module-3

- 1) Sentence Corrections
- 2) Fill the blanks with appropriate words/articles/preposition/verbs/adverbs/conjunction.
- 3) Reading Comprehension (Advance Level)
- 4) Vocabulary

Newspaper reading: The Hindu & Economic Times.

Fifth Semester Syllabus

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	CC	BCA501	Software Engineering	3	0	3	4.5
2	CC	BCA502	Web Programming with PHP	3	0	3	4.5
3	CC	BCA503	Network Programming & System Administration	3	1	3	4.5
4	DE*		Elective-I	3	0	3	4.5
5	DE**		Elective-II	3	0	0	3
6	DE***		Elective-III	3	0	0	3
6	PTI		Minor Project	0	0	3	1.5
7.	GS	GSC501	ESP & SDP-V	0	1	0	1
Total							26.5

Students have to take one mandatory NPTEL course-based on the respective year's offered courses.

Suggestive Choice Based Subjects

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	DE*	BCA504A	Network Security & Cryptography	3	0	3	4.5
2	DE*	BCA504B	Search Engine Optimization	3	0	3	4.5
3	DE**	BCA505A	Mobile Computing	3	0	0	3
4	DE**	BCA505B	Compiler Design	3	0	0	3
5	DE***	BCA506A	Fundamental of Data Analytics	3	0	0	3
6	DE***	BCA506B	Management Information System	3	0	0	3

TITLE OF COURSE: SOFTWARE ENGINEERING

COURSE CODE: BCA501

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and computer architecture.

Introduction:

Software engineering concepts include the collection of tools, procedures, methodologies, and accumulated knowledge about the development and maintenance of software-based systems. This course is strongly suggested for any student planning to take an internship in Computer Science. After an overview of the phases of the software lifecycle, current methodologies, tools, and techniques being applied to each phase will be discussed in depth with localized exercises given to reinforce learning of concepts.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of software development and maintenance. To reach this goal, the following objectives need to be met:

1. Define the current state of software development and maintenance characterized as "the software crisis."
2. Understand the multidimensional aspect of software engineering, which is the current best attempt at solving the software crisis.
3. Become familiar with popular models of the software development and maintenance process.
4. Using the waterfall model, study the inputs, outputs, and processes present in each phase.
5. Study the core concepts present in several popular methodologies and be able to identify strengths and weaknesses of each.
6. Understand the requirement and design approach to develop a software product.
7. Study existing CASE tools to be able to identify opportunities to automate tasks through the use of such tools.
8. Understand the testing techniques to develop an error free software product.
9. Briefly investigate problems present in project management and understand the estimation, techniques during a software development cycle.
10. Consider the issues and techniques present in confidence gaining measures residing in each phase of the software lifecycle.

Course Contents:

Module-1: Interactive Systems, Usability, Introduction to software engineering, Software Process Models, Software life cycle.

Module-2: Requirement Engineering: Requirement Elicitation, Analysis, Specification, SRS, Formal system development techniques.

Module-3: Analysis and Modeling: Data modeling, Functional modeling Software Architecture and Design: Data design, Architectural Design Process, SADT, OOAD, function-oriented design, Design Patterns: Structural Patterns, Behavioral Patterns, and Creational Patterns.

Module-4:- UML: Use case diagram, State chart diagram, Activity Diagram, Class Diagram, Sequence diagram, Collaboration diagram, Deployment Diagram, Event trace diagram.

Module-5: Software Estimation: Software Project Estimating Size, Effort and Cost: Metric for Analysis, Metric for Design, COCOMO model, Putnam Model etc., Implementation and Integration: Coding standard and practices.

Module-6: Software Testing: Top-Down and Bottom-up Approach, Verification and Validation, Structural testing, functional Testing, Testing Strategies, Test Case design.

Module-7: Software Reliability and Quality Management: Total Quality Management, Quality Assurance and Reliability ISO 9000, SEI-CMM, CMMI

Model-8: Software Maintenance: Types, Cost of Software, maintenance, Software Maintenance Models, CASE Tool Taxonomy: Business Process Engineering tool, Process modeling and management tool, project planning tool, requirement tracking tool, Metric and management tool, documentation tool, system software tool etc.

Text Books

1. Software Engineering: A practitioner's approach: Roger S. Pressman, McGraw- Hill Publications (Sixth Edition).
2. Fundamentals of Software Engineering: Mall, Rajib, Prentice Hall of India, New Delhi (2nd Edition).

References

1. Software Testing Techniques, B. Beizer.
2. Structured Systems Analysis: Tools and Techniques, Gane and Sarson.
3. Software Engineering, Sommerville, Addison Wesley.
4. Modern Structured Analysis, E. Yourdon.
5. An Integrated approach to Software Engineering: PankajJalote, NarosaPublishingHouse.
6. Structured design, E. Yourdon and L.Constantine.
7. Fundamentals of Software Engineering: Ghezzi, Jazayeri, Mandriol, PHI

TITLE OF COURSE: SOFTWARE ENGINEERING LAB

COURSE CODE: BCA591

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students must have already registered for the course, “Software Engineering”.

Objectives:

Students will be capable to acquire the generic software development skill through various stages of software life cycle. He will also be able to ensure the quality of software through software development with various protocol based environment.

Course Outcomes (CO):

Students will also be able to handle software development models through rational method. After completion of course student will be able to generate test cases for software testing.

Course Contents:

Module-1: Software Process Model: A Model of Software Development

Module-2: Requirement Analysis: Requirement Elicitation, specification, IEEE standard template for SRS, Requirement Engineering tools.

Module-3: UML Modeling: Use case diagram, State diagram, Activity Diagram, Class Diagram, Sequence diagram, Collaboration diagram, Deployment Diagram, Component Diagram, Event trace diagram, C++ code generation.

Module-4: Software Metrics: Product, process and project metrics.

Module-5: Software Testing: Structural testing, functional Testing, Testing Strategies and Tactics, Test Case design.

List of Experiments

1. Identifying the Requirements from Problem Statements Requirements, Characteristics of Requirements, Categorization of Requirements, Functional Requirements, Identifying Functional Requirements
2. E-R Modeling from the Problem Statements, Entity Relationship Model, Entity Set and Relationship Set, Attributes of Entity, Keys, Weak Entity, Entity Generalization and Specialization, Mapping Cardinalities, ER Diagram, Graphical Notations for ER Diagram Importance of ER modeling

3. Identifying Domain Classes from the Problem Statements, Domain Class, Traditional Techniques for Identification of Classes, Grammatical Approach Using Nouns, Advantages, Disadvantages, Using Generalization, Using Subclasses, Steps to Identify Domain Classes from Problem Statement, Advanced Concepts
4. Modeling UML Use Case Diagrams and Capturing Use Case Scenarios, Use case diagrams, Actor, Use Case, Subject, Graphical Representation, Association between Actors and Use Cases, Use Case Relationships, Include
5. Modeling UML Class Diagrams and Sequence diagrams, Structural and Behavioral aspects, Class diagram, Elements in class diagram, Class Relationships, Sequence diagram, Elements in sequence diagram, Object, Life-line bar, Messages.
6. Modeling Data Flow Diagrams, Data Flow Diagram, Graphical notations for Data Flow Diagram, Explanation of Symbols used in DFD, Context diagram and leveling DFD
7. Statechart and Activity Modeling Statechart Diagrams, Building Blocks of a Statechart Diagram State, Transition, Action, Guidelines for drawing Statechart Diagrams, Activity Diagrams, Components of an Activity Diagram, Activity, Flow Decision, Merge, Fork, Join, Note, Partition, A Simple Example, Guidelines for drawing an Activity Diagram
8. Estimation of Project Metrics Project Estimation Techniques, COCOMO, Basic COCOMO Model, Intermediate COCOMO Model, Complete COCOMO Model, Advantages of COCOMO, Drawbacks of COCOMO, Halstead's Complexity Metrics.
9. Estimation of Test Coverage Metrics and Structural Complexity, Control Flow Graph, Terminologies, McCabe's Cyclomatic Complexity, Computing Cyclomatic Complexity, Optimum Value of Cyclomatic Complexity, Merits, Demerits
10. Ant - Deploying Applications, build.xml, Ant - Executing Java code, Ant - Creating WAR files, basedir, compress, Keepcompression, destfile, duplicate, excludes, excludesfile, includes, includesfile, update.

References

1. R.S. Pressman, "Software Engineering: A Practitioner's Approach", 7Edition, McGraw Hill, 2010
2. Fundamentals of Software Engineering: Mall, Rajib, Prentice Hall of India, New Delhi (2nd Edition).
3. Sommerville, "Introduction to Software Engineering", 8Edition, Addison-Wesley, 2007
4. Ghezzi, Jazayeri and Mandrioli, "Fundamentals of Software Engineering", 2Edition, Prentice-Hall, 2003
5. Peters and Pedrycz, "Software Engineering: An Engineering Approach, John Wiley, 2004
6. Len Bass, "Software Architecture in Practice", 2Edn. Addison Wesley, 2003
7. Allamaraju, "Professional Java Server Programming", Apress, 2004

TITLE OF COURSE: WEB PROGRAMMING WITH PHP

COURSE CODE: BCA502

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Introduction:

This course will enable students to

1. Illustrate the Semantic Structure of HTML and CSS
2. Compose forms and tables using HTML and CSS
3. Design Client-Side programs using JavaScript and Server-Side programs using PHP
4. Infer Object Oriented Programming capabilities of PHP
5. Examine JavaScript frameworks such as jQuery and Backbone

Course Outcomes (CO):

After studying this course, students will be able to

1. Adapt HTML and CSS syntax and semantics to build web pages.
2. Construct and visually format tables and forms using HTML and CSS
3. Develop Client-Side Scripts using JavaScript and Server-Side Scripts using PHP to generate and display the contents dynamically.
4. Appraise the principles of object oriented development using PHP
5. Inspect JavaScript frameworks like jQuery and Backbone which facilitates developer to focus on core features.

Course Contents:

Module-1: Introduction to HTML, What is HTML and Where did it come from?, HTML Syntax, Semantic Markup, Structure of HTML Documents, Quick Tour of HTML Elements, HTML5 Semantic Structure Elements, Introduction to CSS, What is CSS, CSS Syntax, Location of Styles, Selectors, The Cascade: How Styles Interact, The Box Model, CSS Text Styling.

Module-2: HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form Accessibility, Microformats, Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Constructing Multicolumn Layouts, Approaches to CSS Layout, Responsive Design, CSS Frameworks.

Module-3: JavaScript: Client-Side Scripting, What is JavaScript and What can it do?, JavaScript Design Principles, Where does JavaScript Go?, Syntax, JavaScript Objects, The Document Object Model (DOM), JavaScript Events, Forms, Introduction to Server-Side Development with PHP, What is Server-Side Development, A Web Server's Responsibilities, Quick Tour of PHP, Program Control, Functions

Module-4: PHP Arrays and Superglobals, Arrays, \$_GET and \$_POST Superglobal Arrays, \$_SERVER Array, \$_FILES Array, Reading/Writing Files, PHP Classes and Objects, Object-Oriented Overview, Classes and Objects in PHP, Object Oriented Design, Error Handling and Validation, What are Errors and Exceptions?, PHP Error Reporting, PHP Error and Exception Handling

Module-5: Managing State, The Problem of State in Web Applications, Passing Information via Query Strings, Passing Information via the URL Path, Cookies, Serialization, Session State, HTML5 Web Storage, Caching, Advanced JavaScript and jQuery, JavaScript Pseudo-Classes, jQuery Foundations, AJAX, Asynchronous File Transmission, Animation, Backbone MVC Frameworks, XML Processing and Web Services, XML Processing, JSON, Overview of Web Services.

Text Books

1. Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", 1st Edition, Pearson Education India.

References

1. Robin Nixon, "Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5", 4th Edition, O'Reilly Publications, 2015. (ISBN:978-9352130153)
2. Luke Welling, Laura Thomson, "PHP and MySQL Web Development", 5th Edition, Pearson Education, 2016. (ISBN:978-9332582736)
3. Nicholas C Zakas, "Professional JavaScript for Web Developers", 3rd Edition, Wrox/Wiley India, 2012. (ISBN:978-8126535088)
4. David Sawyer Mcfarland, "JavaScript & jQuery: The Missing Manual", 1st Edition, O'Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014 (ISBN:978-9351108078)
5. Zak Ruvalcaba Anne Boehm, "Murach's HTML5 and CSS3", 3rd Edition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016.

TITLE OF COURSE: WEB PROGRAMMING WITH PHP LAB

COURSE CODE: BCA592

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Objectives:

To learn and understand Web design. It is a process of conceptualizing, planning, and building a collection of electronic files that determine the layout, color, text styles, structure, graphics, images, and use of interactive features that deliver pages to your site.

Course Outcomes (CO):

1. Apply critical thinking and problem solving skills required to successfully design and implement a web site.
2. Demonstrate the ability to analyze, identify and define the technology required to build and implement a web site.
3. Apply critical thinking and problem solving skills required to successfully design and implement a web site.
4. Demonstrate the ability to analyse, identify and define the technology required to build and implement a web site.
5. Demonstrate knowledge of artistic and design components that are used in the creation of a web site.
6. Utilize and apply the technical, ethical and interpersonal skills needed to function in a cooperative environment.

Course Contents:

1. Concepts of networking, Web and HTML. Introduction with Web, Network, Website, Server, Client side, Server side and other terms related to basic website designing concept. Introduction with HTML language.
2. Introduction with .net. Introduction of Microsoft .net, Explain features and phases of the object-oriented approach. (C#) Basic Syntax, Reading and writing to a console, Data Types, Type Conversion, Variables, Constants.
3. SQL Server, Introduction with SQL Server, Role of a Database Server, SQL language, Working With Database (Table concepts), SQL query (Data Definition Language, Data Manipulation Language, Data Control Language)
4. Hands on C# language. Introduction of C# and programing basic of C#, Programs on different problems in C#. Introduction with core PHP, Core PHP introduction and programing concepts. Quick start with PHP programs. Database Integration in PHP.

Exercises that must be done in this course are listed below:

Exercise No.1: Create a form in HTML for entering value for some specific fields. (Registration Page)

Exercise No.2: Create table in SQL for storing data of registration page. (Using sql query)

Exercise No.3: Create a webpage to show the data which is entered in sql tables through registration page.

Exercise No. 4: Create a web page to file upload option, so user can upload document on website.

Exercise No. 5: Create a webpage to show the uploaded document.

Exercise No. 6: assemble all the web page to create a website for a specific organization. (Minor project).

Exercise No. 7: Create master page for previous developed pages.

Exercise No. 8: Apply validators for all fields which are used in previous developed pages.

Exercise No. 9: Major project.

Text Book:

3. Learning Web Design, Book by Jennifer Niederst Robbins
4. Professional PHP 6, **Publisher:** Wiley

Recommended Systems/Software Requirements:

1. Desktop PC with minimum of 166 MHZ or faster processor with at least 1 GB RAM and 160 GB disk space.
2. Visual studio 2012, Microsoft sql server 2008 R2

TITLE OF COURSE: NETWORK PROGRAMMING & SYSTEM ADMINISTRATION

COURSE CODE: BCA503

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and computer architecture.

Introduction:

The Internet transfers PettaBytes of data every minute across the globe. Do you have any idea how that happens? You click on a link and you download a multi MegaByte(MB) file within seconds, watch your favourite videos, do live video calls and what not. This is not magic! There

are people behind the scene, who are working very hard to make all this happen, both efficiently and safely, by making sure that the data reaching you is unsniffed and legitimate. They are called Security Engineers. All some of you might be wondering about what are you going to learn here. Well if you have learnt programming you must have wondered, Ok, we know programming! But, what to do with it? What's its practical implementation. After all, no one is going to give you an array and will ask you to sort it or so. Right! *Real world problems that are solved by programming* are different and complicated

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of Network Programming. To reach this goal, the following objectives need to be met:

1. To master the terminology and concepts of the OSI reference model and the TCP-IP reference model.
2. To master the concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks,
3. To be familiar with wireless networking concepts,
4. To be familiar with contemporary issues in networking technologies,
5. To be familiar with network tools and network programming

Course Contents:

Module-1: History / OSI model; Unix commands, High level UDP and TCP/IP; UDP sockets, tftp; Sorcerer's Apprentice

Module-2: Linux basics, C system calls, fork, Signals, Byte ordering, misc. inet functions

Module-3: TCP sockets (SOCK_STREAM); Client / server model; Daemons, Fall 2018 Career Fair - lecture canceled

Module-4: select (I/O Multiplexing); Bonjour (Zeroconf) / Service Discovery, TCP 3 Way Handshake, TCP States, TCP Close, Non-Blocking I/O, TCP congestion control, ACK, windows, ECN

Module-5: Threads, Application Layer Protocols; Telnet; FTP, IRC,

Module-6: Broadcasting, IPv4/IPv6 addressing, DNS, gethost*(), nslookup, Bit Torrent/ DHT / P2P

Module-7: Socket Options, Out of Band Sockets

Module-8: HTTP, Security Basics, Security Basics II

Module-9: SSL, RPISEC Guest Lecture

Module-10: gRPC, IPFS / QUIC, SDNs/ OpenFlow / Mininet, Buffer Bloat

Text Books

1. The Practice of System and Network Administration, Second Edition Thomas A. Limoncelli, Christina J. Hogan , Strata R. Chalup

References

1. Advanced Linux Networking, Roderick W. Smith, Addison-Wesley Professional (Pearson Education), 2002.

2. Linux Network Administrator's Guide, Tony Bautts, Terry Dawson, Gregor N. Purdy, O'Reilly, Third Edition, 2005

Prerequisite: Computer Networking Course

TITLE OF COURSE: NETWORK PROGRAMMING & SYSTEM ADMINISTRATION LAB

COURSE CODE: BCA593

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Students must have already registered for the course, basic computation, linux.

Objectives: This course provides an intensely practical introduction to basic Unix Systems Management, which involves both administration and programming. The Linux Operating System will be used extensively throughout this course for lecture/tutorial examples and importantly during laboratory classes. In addition to system administration, TCP/IP network service administration and the use of scripting languages will be introduced.

Course Outcomes: This course is an option course so it is not required to contribute to the development of program learning outcomes (PLOs) though it may assist your achievement of several PLOs.

Course Contents:

This course will serve to broaden the student's understanding of the issues and latest developments in the area of Network Programming.

1. To master the terminology and concepts of the OSI reference model and the TCP-IPreference model.

2. To master the concepts of protocols, network interfaces, and design/performance issues in local area networks and wide area networks,
3. To be familiar with wireless networking concepts
4. To be familiar with contemporary issues in networking technologies
5. To be familiar with network tools and network programming

Course Contents:

Module-1: Server/Client Installation over VMware Environment

Module-1: Packet Analysis by using TCPDUMP and WIRESHARK

Module-2: Network Practice with Packet Tracer

Module-3: System Administration: User/Group management, File System Management

Module-4: Network Configuration: Start/Stop network Service, network interface configuration

Module-5: Firewall Configuration

Module-6: DNS and DHCP Configuration and Troubleshooting

Module-7: Web and Proxy Server Configuration and Troubleshooting

Module-8: Basic Mail Server Configuration and Troubleshooting

Module-9: SAMBA, NFS, CUPS and FTP configuration and Troubleshooting

Module-10: Webmin/SSH configuration

References

1. The Practice of System and Network Administration, Second Edition Thomas A. Limoncelli, Christina J. Hogan , Strata R. Chalup
2. Advanced Linux Networking, Roderick W. Smith, Addison-Wesley Professional (Pearson Education), 2002.
3. Linux Network Administrator's Guide, Tony Bautts, Terry Dawson, Gregor N. Purdy, O'Reilly, Third Edition, 2005

TITLE OF COURSE: NETWORK SECURITY & CRYPTOGRAPHY

COURSE CODE: BCA504A

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Pre-requisite: Participants will be expected to have a fairly good background in discrete mathematics. Since the tools used in this course will be based on Linux/FOSS, participants will need to be comfortable working with Linux/FOSS.

Introduction:

Cryptography provides important tools for ensuring the privacy, authenticity, and integrity of the increasingly sensitive information involved in modern digital systems. Nowadays, core cryptographic tools, including encryption, message authentication codes, digital signature, key agreement protocols, etc., are used behind millions of daily on-line transactions. In this course, we will unveil some of the "magic" of cryptography. Modern Cryptography uses mathematical language to precisely pin down elusive security goals, design primitives and protocols to achieve these goals, and validate the security of designed primitives and protocols using mathematical proofs based on clearly stated hardness assumptions. Therefore, to learn cryptography, it is essential to understand its mathematical underpinning. In this class, we will see the inner-working of cryptography for several core cryptographic tools, from encryption, to message authentication codes, to hash functions, to digital signatures, etc.

Course Outcomes (CO):

This subject focuses on the foundation of cryptography. It is not about learning all cryptographic acronyms or all cryptographic protocols in use today, nor about familiarizing with current implementations of cryptographic tools, and issues that arise when implementing cryptographic tools. Furthermore, this class is certainly not going to cover the more extensive topic of computer security. In particular, this course will not teach you about firewalls, malware detection, nor how to secure or hack a system. Rather, the course will strive to convey the ideas and principles behind cryptographic design, and cryptographic applications. To reach this goal, the following objectives need to be met:

1. It gives thorough knowledge about the computer security (what is computer security, types of attacks on it, protection methods etc.)
2. Understand the actual working of OSI model in networking.
3. Explore the student's knowledge about the how to transmit data from source to destination.
4. Students will get to know about what is cryptography, their elements, taxonomy of techniques etc.
5. It focused on symmetric key encryption algorithm. It will give comprehensive acquaintance about symmetric key approach.

6. Students will be able to understand the asymmetric key encryption algorithm as RSA, SHA etc.
7. Understand the
8. Discuss about the IP protocols a SSL. It will also focus on basics of authentication, and password authentication tokens etc.
9. Student will able to learn about Email security, S/MIME etc.
10. The primary objective of this course is to demonstrates suitable knowledge to students about the firewall, their attacks etc.

Course Contents:

Module-1: I Introduction: Security Goals, Cryptographic Attacks, Services and Mechanism, Techniques. Mathematics of Cryptography: Integer Arithmetic, Modular Arithmetic, Matrices, Linear Congruence.

Module-2: Traditional Symmetric-Key Ciphers: Introduction, Substitution Ciphers, Transpositional Ciphers, Stream and Block Ciphers. Data Encryption Standard (DES): Introduction, DES Structure, DES Analysis, Security of DES, Multiple DES, Examples of Block Ciphers influenced by DES. Advanced Encryption Standard: Introduction, Transformations, Key Expansion, The AES Ciphers, Examples, Analysis of AES.

Module-3: Encipherment using Modern Symmetric-Key Ciphers: Use of Modern Block Ciphers, Use of Stream Ciphers, Other Issues. Mathematics of Asymmetric-Key Cryptography: Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation and Logarithm. Asymmetric Key Cryptography: Introduction, RSA Cryptosystem, Rabin Cryptosystem, Elgamal Cryptosystem, Elliptic Curve Cryptosystems.

Module-4: Cryptography Hash Functions: Introduction, Description of MD Hash Family, Whirlpool, SHA-512. Digital Signature: Comparison, Process, Services, Attacks on Digital Signature, Digital Signature Schemes, Variations and Applications. Key Management: Symmetric-Key Distribution, Kerberos, Symmetric-Key Agreement, Public-Key Distribution, Hijacking.

Module-5: Security at the Application Layer: PGP and S/MIME: Email, PGP, S/MIME. Security at the Transport Layer: SSL and TLS: SSL Architecture, Four Protocols, SSL Message Formats, Transport Layer Security. Security at the Network Layer: IPSec: Two modes, Two security protocols, Security association, security policy, Internet Key exchange, ISAKMP.

Text Books

1. B. A. Forouzan – “Data Communications and Networking (3rd Ed.)”- TMH
2. A. S. Tanenbaum – “Computer Networks (4th Ed.)” – Pearson Education/PHI
3. W. Stallings – “Data and Computer Communications (5th Ed.)” – PHI/ Pearson Education
4. Zheng&Akhtar, Network for Computer Scientists & Engineers, OUP
5. Black, Data & Computer Communication, PHI
6. Miller, data Communication & Network, Vikas

Reference Books

1. Kurose and Rose – “ Computer Networking -A top down approach featuring the internet” – Pearson Education
2. Leon, Garica, Widjaja – “Communication Networks” – TMH
3. Walrand – “Communication Networks” – TMH.
4. Comer – “Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.)” – Pearson Education/PHI
1. Software Engineering: A practitioner’s approach: Roger S. Pressman, McGraw- Hill Publications (Sixth Edition).
2. Fundamentals of Software Engineering: Mall, Rajib, Prentice Hall of India, New Delhi (2nd Edition).

TITLE OF COURSE: NETWORK SECURITY & CRYPTOGRAPHY LAB

COURSE CODE: BCA594A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Participants will be expected to have a fairly good background in discrete mathematics

Objectives: Students will be capable to acquire the generic practical and theoretical knowledge of cryptography and network security

Course Outcomes (CO):

At the end of the course, the student should be able to:

1. Implement the cipher techniques
2. Develop the various security algorithms
3. Use different open source tools for network security and analysis

Course Contents:

Module-1: Cryptography & Information Security Introduction: Classical Encryption Schemes, Principles of Modern Cryptography, Perfect Secrecy and Its Limitations, Attacks on Computers & Computer Security, Need for Security, Security approaches, Principles of Security, Types of attack.

Module-2: Cryptography Concepts & Techniques: Introduction, Plaintext & Cipher text, Substitution Techniques, Transposition Techniques, Encryption & Decryption, Symmetric & Asymmetric key Cryptography, Key Range & Key Size

Module-3: Symmetric Key Algorithm: Introduction, Algorithm types & Modes: Overview of Symmetric Key Cryptography, Computational Security, Pseudorandom Generators and Stream Ciphers,

Module-4: Asymmetric Key Algorithm: Public-Key Encryption: Definitions of Security, Hybrid Encryption and the KEM/DEM Paradigm, Digital Signature and RSA Introduction: Overview of Asymmetric key Cryptography, RSA algorithm, Symmetric & Asymmetric key

Module-5: Internet Security Protocols: User Authentication Basic Concepts, SSL protocol, Authentication Basics, Password, Authentication Token, Certificate based Authentication, Biometric Authentication.

Module-6: Electronic Mail Security: Basics of mail security, Pretty Good Privacy, S/MIME.

Module-7: Firewall: Introduction, Types of firewall, Firewall Configurations, DMZ Network

List of Experiments

1. Write a C program that contains a string (char pointer) with a value 'Hello World'. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello World'. The program should AND or and XOR each character in this string with 127 and display the result
3. Write a Java program to perform encryption and decryption using the following algorithms: a) Ceaser Cipher b) Substitution Cipher c) Hill Cipher
4. Write a Java program to implement the DES algorithm logic
5. Write a C/JAVA program to implement the Blow Fish algorithm logic
6. Write a C/JAVA program to implement the Rijndael algorithm logic
7. Using Java Cryptography, encrypt the text "Hello world" using Blow Fish. Create your own key using Java keytool.
8. Write a Java program to implement RSA Algorithm
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob).
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the SHA-1 algorithm in JAVA. Write a Java program to implement RSA Algorithm

References

1. Kurose and Rose – "Computer networking -A top down approach featuring the internet" – Pearson Education

2. Leon, Garica, Widjaja – “Communication Networks” – TMH
3. Walrand – “Communication Networks” – TMH.
4. Comer – “Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.)” – Pearson Education/PHI

TITLE OF COURSE: SEARCH ENGINE OPTIMIZATION

COURSE CODE: BCA504B

L-T-P: 3-0-3

CREDITS: 4.5

SYLLABUS

Introduction:

Ever wonder how major search engines such as Google, Bing and Yahoo rank your website within their searches? Or how content such as videos or local listings are shown and ranked based on what the search engine considers most relevant to users? Welcome to the world of Search Engine Optimization (SEO). This course is the first within the SEO Specialization and it is intended to give you a taste of SEO.

Course Outcomes (CO):

The objectives of this course is:

- To introduce the foundational elements of how search engines work, how the SEO landscape has changed and what you can expect in the future.
- To discuss core SEO strategies and tactics used to drive more organic search results to a specific website or set of websites, as well as tactics to avoid to prevent penalization from search engines.
- To discover how to position yourself for a successful career in SEO should this subject prove interesting to you? We hope this taste of SEO, will entice you to continue through the Specialization.

Course Outcomes (CO):

Students will be able to learn about to:

1. Critique the role of advertisements and corporate funding in the development of search
2. Write your own content for a website that will improve search results
3. Compare and contrast the functionality of search engine algorithms updates
4. Develop an optimization strategy following best practices for a client to implement to help increase their ranking

Course Contents:**Module-1: Getting Started & Introduction to SEO**

Introduction to SEO, How I Got Started in SEO, Role of an SEO Manager.

Module-2: Evolution of SEO

Evolution of SEO, How Search Engines Work, History of Search Engines (1945 - 1993), Advancement of Search (1993 - 2000)

Module-3: Current SEO Best Practices

Introduction to Search Engine Algorithms, SEO Best Practices and Ranking Factor, Early Algorithms: Florida, Austin, and Brandy, Updates: Personalized, Universal, Vince, and Caffeine, Panda: The Game Changer for Content, Cleaning Up Links with Penguin, Refinements: Hummingbird, Pigeon, and Mobile, How You Can Avoid Penalties

Module-4: SEO of Today, Tomorrow and Beyond & Course Wrap-Up

SEO of Today, Tomorrow, and Beyond, Evolution of Keyword Optimization, Strengthening Your Keyword Strategy, How Does Branding Influence Website Rank?

Books:

1. Search Engine Optimization All-in-One For Dummies 3rd Edition by Bruce Clay

Reference Books:

1. SEO 2016: Learn Search Engine Optimization (SEO Books Series Book 1) by R.L Adams

TITLE OF COURSE: SEARCH ENGINE OPTIMIZATION LAB

COURSE CODE: BCA594B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Prerequisite: Knowledge on Internet, HTML, CSS, XML.

Introduction:

Ever wonder how major search engines such as Google, Bing and Yahoo rank your website within their searches? Or how content such as videos or local listings are shown and ranked

based on what the search engine considers most relevant to users? Welcome to the world of Search Engine Optimization (SEO). This course is the first within the SEO Specialization and it is intended to give you a taste of SEO.

Course Objective:

The objectives of this course is:

- To introduce the foundational elements of how search engines work, how the SEO landscape has changed and what you can expect in the future.
- To discuss core SEO strategies and tactics used to drive more organic search results to a specific website or set of websites, as well as tactics to avoid to prevent penalization from search engines.
- To discover how to position yourself for a successful career in SEO should this subject prove interesting to you? We hope this taste of SEO, will entice you to continue through the Specialization.

Tools Requirements:

- Webmasters tools
- Analytics
- Keyword Planner
- SEO Marketing Tools
- Website Analysis Tools
- Trend Analysis Tools
- Related keyword research tools

Course Content:

Lab-1: SEO analysis of website

- 1.1 Website Analysis
- 1.2 SEO Competition Analysis
- 1.3 Keyword Research
- 1.4 Keyword Finalize
- 1.5 Search Engine Optimization Strategy Analysis
- 1.6 Initial Ranking analysis

Lab-2: On Page Search Engine Optimization

- 1.1 Website Structure Optimization
- 1.2 Content Optimization
- 1.3 Image Optimization
- 1.4 HTML Code Optimization
- 1.5 Web master tools

1.6 RSS Feed Generation

Lab-3: OFF PAGE SEO OPTIMIZATION

2.1 Google Base Optimization (Google My Business)

2.2 Reputation building

2.3 Book marking

2.4 SEO Content writing

2.5 Quality link building

2.6 Article submission

2.7 Press Release Creation

2.8 Blog commenting

2.9 Forum posting

Lab-4: Social media optimization & marketing

3.1 Social bookmarking

3.2 Social Networking Profiles creations

3.4 Social Networking Profile Optimizations

3.5 Content Sharing Optimization

3.6 Blogging

3.7 Blog Optimization

Lab-5: Search engine marketing

4.1 Slide submission

4.2 Pay per click (PPC) campaigns

4.3 Search Engine Marketing Research

4.4 Video Marketing

4.5 Image marketing

Lab-6: Reputation building

5.1 Authority building

5.2 Trust building

5.3 Relationship building

5.4 Mentions

5.5 Tagging

5.6 Profile optimization

5.7 Profile building

Lab-7: Conversion optimization

- 6.1 Conversion analysis
- 6.2 Conversion Tracking
- 6.3 Conversion Creation
- 6.4 Conversion building
- 6.5 CTR Improvements tactics

Books:

1. Search Engine Optimization All-in-One for Dummies 3rd Edition by Bruce Clay
2. SEO 2016: Learn Search Engine Optimization (SEO Books Series Book 1) by R.L Adams

TITLE OF COURSE: MOBILE COMPUTING

COURSE CODE: BCA505A

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Familiarity with concepts of computer networks.

Introduction:

The course deals with basic concepts of mobile computing and will equip students with understanding of network protocol stack, Adhoc networks and different mobile platforms.

Course Outcomes (CO):

The course will teach the concepts of mobile networking and basics of mobile communication systems. To reach this goal, the following objectives need to be met:

1. Explain the basics of mobile communication system.
2. Illustrate the generations of telecommunication systems in wireless networks.
3. Organize the functionalities and components of mobile computing systems into different layers and apply various techniques for realizing the functionalities.
4. Be able to choose the functionality required at each layer of application.
5. Be able to determine the functionality of MAC, network layer and Identify a routing protocol for a given Ad hoc network.
6. Explain the functionality of Transport and Application layers.

7. Develop skills of finding solutions and building software for mobile computing applications.
8. Develop mobile computing applications by analyzing their characteristics and requirements, selecting the appropriate computing models and software architectures, and applying standard programming languages and tools

Course Contents:

Module-1: Introduction to Mobile Computing: Characteristics of Mobile Computing, Applications of Mobile Computing, Mobile Computing vs Wireless Networking, MAC protocols, Spread Spectrum, Multiplexing.

Module-2: Mobile Telecommunication Systems: GSM service and Architecture, Connection establishment, Frequency allocation, Routing, Mobility management, Security, GPRS, UMTS architecture, handover, security.

Module-3: Mobile Network Layer: Mobile IP, DHCP, Adhoc, Proactive Protocol-DSDV, Reactive Routing Protocol-DSR, AODV, Multicast Routing-ODMRP, Vehicular Ad Hoc networks, MANET vs VANET.

Module-4: Mobile Transport and Application Layer: Mobile TCP,WAP Architecture, WDP and WTLS, WTP, WSP , WAE , WTA Architecture ,WML.

Module-5: Mobile Platform and Applications : Mobile Device Operating Systems, Special Constraints & Requirements ,Commercial Mobile Operating Systems , Software Development Kit: iOS, Android, BlackBerry, Windows Phone , MCommerce – Structure , Pros & Cons, Mobile Payment System,Security Issues

Text Books

1. Jochen Schiller – "Mobile Communications" – 2nd Ed. – PHI.
2. Prasant Kumar Pattnaik, Rajib Mall – "Fundamentals Of Mobile Computing" – 2nd Ed. – PHI Learning.

References

1. Dharma Prakash Agarval, Qing and An Zeng – "Introduction to Wireless and Mobile systems" – Thomson Asia
2. UweHansmann, LotharMerk, Martin S. Nicklons and Thomas Stober–"Principles of Mobile Computing" – Springer
3. William.C.Y.Lee–"Mobile Cellular Telecommunications-Analog and Digital Systems" – Tata McGraw Hill
4. C.K.Toth – "AdHoc Mobile Wireless Networks" – Pearson Education

TITLE OF COURSE: COMPILER DESIGN

COURSE CODE: BCA505B

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages, and computer architecture.

Introduction:

This course examines compiler design concepts, phases of compiler in detail and cousins of compiler. The Topics to be covered (tentatively) include:

- Introduction to Compiler
- Lexical Analysis
- Syntax Analysis
- Type Checking
- Intermediate Code Generation
- Code Generation
- Code Optimization.

Course Outcomes (CO):

After completion of the course, students will able:

1. To apply the knowledge of lex tool & yacc tool to develop a scanner & parser.
2. To design & conduct experiments for Intermediate Code Generation in compiler.
3. To design & implement a software system for backend of the compiler.
4. To deal with different translators.
5. To develop program to solve complex problems in compiler
6. To learn the new code optimization techniques to improve the performance of a program in terms of speed & space.
7. To acquire the knowledge of modern compiler & its features.
8. To learn & use the new tools and technologies used for designing a compiler
9. To use the knowledge of patterns, tokens & regular expressions for solving a problem in the field of data mining.
10. To apply compiler design techniques.
11. To relate theory of compiler design to practice
12. To learn to build domain specific generators

Course Contents:

Module-1: Introduction to Compiling

Compilers, Analysis-synthesis model, the phases of the compiler, Cousins of the compiler.

Lexical Analysis

The role of the lexical analyzer, Tokens, Patterns, Lexemes, Input buffering, Specifications of a token, Recognition of tokens, Finite automata, From a regular expression to an NFA, From a regular expression to NFA, From a regular expression to DFA, Design of a lexical analyzer generator (Lex).

Module-2: Syntax Analysis

The role of a parser, Context free grammars, Writing a grammar, Top down Parsing, Non-recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error Recovery strategies for different parsing techniques.

Syntax directed translation

Syntax directed definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes

Module-3: Type checking

Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions

Run time environments

Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies, Parameter passing (call by value, call by reference, copy restore, call by name), Symbol tables, dynamic storage allocation techniques.

Module-4: Intermediate code generation

Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).

Code optimization

Introduction, Basic blocks & flow graphs, Transformation of basic blocks, Dag representation of basic blocks, The principle sources of optimization, Loops in flow graph, Peephole optimization.

Code generations Issues in the design of code generator, a simple code generator, Register allocation & assignment

Module-5: Software Testing: Top-Down and Bottom-up Approach, Verification and Validation, Structural testing, functional Testing, Testing Strategies, Test Case design.

Text Books

1. Aho, Sethi, Ullman - "Compiler Principles, Techniques and Tools" - Pearson Education.
2. Holub - "Compiler Design in C" – PHI 3. Tremblay and Sorenson Compiler Writing-McgrawHill International.
3. Chattopadhyay , S- Compiler Design (PHI)

References

1. J. Archer Harris, Operating systems – Schuam's outlines, Tata McGraw Hill.
2. Gary Nutt, Operating Systems – A modern perspective, Pearson Education.

TITLE OF COURSE: FUNDAMENTAL OF DATA ANALYTICS

COURSE CODE: BCA506A

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: To extract the maximum from the course, the following prerequisites are must.

- A strong mathematical background in Probability and Statistics
- Proficiency with algorithms
- Programming skills in C, Python, R, etc.
- Critical thinking and problem solving skills

Introduction: This course will cover fundamental algorithms and techniques used in Data Analytics. The statistical foundations will be covered first, followed by various machine learning and data mining algorithms. Technological aspects like data management (Hadoop), scalable computation (Map-Reduce) and visualization will also be covered. In summary, this course will provide exposure to theory as well as practical systems and software used in data analytics.

After completing this course, you will learn how to:

- Find a meaningful pattern in data
- Graphically interpret data
- Implement the analytic algorithms
- Handle large scale analytics projects from various domains
- Develop intelligent decision support systems

Course Outcomes (CO):

Upon completion of the course, the students will be able to:

1. Work with big data tools and its analysis techniques
2. Analyze data by utilizing clustering and classification algorithms

3. Learn and apply different mining algorithms and recommendation systems for large volumes of data
4. Perform analytics on data streams
5. Learn No-SQL databases and management.

Course Contents:

Module-1: INTRODUCTION TO BIG DATA:

Evolution of Big data, Best Practices for Big data Analytics, Big data characteristics, Validating, The Promotion of the Value of Big Data, Big Data Use Cases - Characteristics of Big Data Applications, Perception and Quantification of Value, Understanding Big Data Storage, A General Overview of High-Performance Architecture, HDFS – Map Reduce and YARN – Map Reduce Programming Model

Module-2: CLUSTERING AND CLASSIFICATION:

Advanced Analytical Theory and Methods: Overview of Clustering, K-means, Use Cases – Overview of the Method, Determining the Number of Clusters, Diagnostics – Reasons to Choose and Cautions, Classification: Decision Trees – Overview of a Decision Tree, The General Algorithm, Decision Tree Algorithms, Evaluating a Decision Tree, Decision Trees in R- Naïve Bayes, Bayes’ Theorem – Naïve Bayes Classifier.

Module-3: ASSOCIATION AND RECOMMENDATION SYSTEM:

Advanced Analytical Theory and Methods: Association Rules, Overview, Apriori Algorithm – Evaluation of Candidate Rules, Applications of Association Rules, Finding Association& finding similarity, Recommendation System: Collaborative Recommendation- Content Based Recommendation, Knowledge Based Recommendation, Hybrid Recommendation Approaches.

Module-4: STREAM MEMORY

Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating moments, Counting oneness in a Window, Decaying Window, Real time Analytics Platform(RTAP) applications, Case Studies – Real Time Sentiment Analysis, Stock Market Predictions. Using Graph Analytics for Big Data: Graph Analytics

Module-5: NOSQL DATA MANAGEMENT FOR BIG DATA AND VISUALIZATION

NoSQL Databases : Schema-less Models: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores, Tabular Stores, Object Data Stores, Graph Databases Hive, Sharding, Hbase, Analyzing big data with twitter, Big data for E-Commerce Big data for blogs Review of Basic Data Analytic Methods using R.

Text Books

1. AnandRajaraman and Jeffrey David Ullman, “Mining of Massive Datasets”, Cambridge University Press, 2012.

References

1. David Loshin, “Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph”, Morgan Kaufmann/Elsevier Publishers, 2013.
2. Software for Data Analysis: Programming with R (Statistics and Computing), John M. Chambers, Springer.

TITLE OF COURSE: MANAGEMENT INFORMATION SYSTEM

COURSE CODE: BCA506B

L-T-P: 3-0-0

CREDITS: 3

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in connection between information systems (IS) and business performance.

Introduction:

The use of information and communication technologies (ICT) by individuals and organizations dominates the business world. There is a fundamental change going on in the way that organizations run businesses and interact with each other. New types of infrastructure and applications are developed and utilized such as ERP (enterprise resource planning), IOS (inter-organizational systems), RFID (radio frequency identification), CRM (customer relationship management), to name a few. The aim of the course is to enable students to assess the opportunities and problems that managers in a wide range of organizations face as they attempt to use these IT applications to add value to their businesses. It also aims to help students understand transformational changes within and across industries. These changes have strategic implications for many businesses.

Course Outcomes (CO):

This course will serve a fundamental change going on in the way that organizations run business and interact with each other:

At the end of the course, you will be able to:

1. Explain basic concepts for IT/IS management
2. Discuss organizational, business and strategic issues surrounding IT/IS, and
3. Analyze and evaluate uses of strategic IT/IS in practice.
4. Understand and articulate fundamental concepts of information technology management.
5. Assess and apply IT to solve common business problems.

6. Suggest and defend effective solutions to business problems, and design a database application to solve a business problem.
7. Discuss the ethical aspects of information technology use in the organization and its governance issues.

Course Contents:

Module-1 Management Information System (MIS): Definition, Characteristics, Subsystems of MIS (Activity and Functional subsystems), Structure of MIS; Reasons for failure of MIS. Understanding Major Functional Systems: Marketing & Sales Systems, Finance & Accounting Systems, Manufacturing & Production Systems, Human Resource Systems, Inventory Systems; their sub systems, description and organizational levels.

Module-2: Decision Support Systems (DSS): Definition, Relationship with MIS, Evolution of DSS, Characteristics, classification, objectives, components, applications of DSS.

Module-3: ERP, CRM, SCM:

ERP (Enterprise Resource Planning):

- a) Concepts of ERP, architecture of ERP, Generic modules of ERP, Applications of ERP, concept of XRP (extended ERP)
- b) Features of commercial software like SAP, Oracle Apps, MS Dynamics NAV, People soft.

CRM (Customer Relationship Management)

- a) Concepts of CRM, Features of CRM (acquisition and retention)
- b) Features of commercial software like I2-Rhythm, SIEBEL

SCM (supply Chain Management):

- a) Concepts of SCM, drivers of SCM, inbound & outbound
- b) Definition, brief description and applicability of: e-Procurement, e-Tailing, e-Logistics, e-collaboration, e-integration.
- c) Case studies for ERP, CRM, SCM

Module-4: Database Management Systems (DBMS) [e.g. MS-Access/ Oracle/ MS SQL Server / My SQL etc.]

- a) What is a DBMS. Need for using DBMS. Concepts of tables, records, attributes, keys, integrity constraints, 3-schema architecture, data independence.
- b) SQL: DDL & DML concepts, SQL commands [ANSI standard].

Module-5: Organization and Computer Networks

Introduction, Basics of computer systems, Basic Network Terminologies, Definitions and Application, The Intranet and the Extranet.

Module-6: Data Warehousing and Data Mining

Concepts of Data warehousing, data mart, meta data, multidimensional modeling, Online Analytical Processing (OLAP), Online Transaction Processing (OLTP), Knowledge Management System (KMS), Active Knowledge Management Server (AKMS) Features of commercial software like Informatics Data mining concepts, knowledge discovery v. data mining, data mining applications. Case studies on data warehousing / data mining

Text Books

1. Dan W. Paterson, Introduction to Artificial Intelligence & Expert System, PHI./Pearson Education.
2. Davis & Olson, Management Information System, Tata McGraw Hill
3. Kroenke, D. M., Gemino, A., & Tingling, P. (2016). *Experiencing MIS* (4th Canadian Edition). Toronto: Pearson. ISBN-13: 9780134078434

References

1. ISRD, Introduction to Database Management Systems, Tata McGraw Hill
2. Ivan Bayross : SQL & PL/SQL , BPB
3. Leon : ERP, Leon Tech Press
4. Loney & Koch: The Oracle 9i Complete Reference, Oracle Press
5. Mahadeo Jaiswal & Monica Mittal : Management Information Systems, OUP

TITLE OF COURSE: ESP & SDP - V

COURSE CODE: GSC501

L-T-P: 0-1-0

CREDITS: 1

Introduction:

The course is on GATE exam preparation, logical reasoning, English sentence correction, English Grammar correction etc.

Course contains:**Module-1**

Stream wise GATE & IES syllabus.

SKILL DEVELOPMENT FOR PROFESSIONALS IV**Module-1**

Quantitative Aptitude & Data Interpretation- Miscellaneous

Module-2**Logical Reasoning**

- 1) Statement and Assumption,
- 2) Statement and Conclusion,
- 3) Statement and Course of Action,
- 4) Cause and Effect,
- 5) Drawing Inference

Module-3

- 1) Sentence Corrections

- 2) Fill the blanks with appropriate words/articles/preposition/verbs/adverbs/conjunction.
- 3) Reading Comprehension (Advance Level)
- 4) Vocabulary

Newspaper reading: The Hindu & Economic Times.

Sixth Semester Syllabus

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	CC	BCA601	Artificial Intelligence	3	1	3	5.5
2	DE*		Elective-I	3	1	0	4
3	DE**		Elective-II	0	0	3	1.5
4	PTI	BCA692	Viva-Voce	0	0	3	3
5	PTI	BCA693	Major Project	0	0	6	6
6	PTI	BCA694	Seminar/Internship	0	0	6	3
7.	GS	GSC601	ESP & SDP-VI	0	1	0	1
Total							24

Students have to take one mandatory NPTEL course-based on the respective year's offered courses.

#Students will undergo project/training/internship in the industry / research organization / reputed Institute during the vacation

Suggestive Choice Based Subjects

Sl No.	Type	Subject Code	Topic	L	T	P	Credit Points
1	DE*	BCA602A	Cloud Computing	3	1	0	4
2	DE*	BCA602B	Distributed System	3	1	0	4
3	DE*	BCA602C	Software Project Management	3	1	0	4
4	DE*	BCA602D	Real Time Operating System	3	1	0	4
5	DE**	BCA693A	Android Programming	0	0	3	1.5
6	DE**	BCA693B	Administration of SQL Server	0	0	3	1.5

TITLE OF COURSE: ARTIFICIAL INTELLIGENCE

COURSE CODE: BCA601

L-T-P: 3-1-3

CREDITS: 5.5

SYLLABUS

Objectives: In this course we will study the basic components of an intelligent system, their functions, mechanisms, policies and techniques used in their implementation and examples.

Course Outcomes (CO):

Students will have a detailed knowledge of the concepts of artificial intelligence, various applications of AI in different fields, Aware of a variety of approaches to AI techniques

Course Contents:

Unit-1 (Introduction to AI): Definitions, Goals of AI, AI Approaches, AI Techniques, Branches of AI, Applications of AI. Introduction of Intelligent Systems: Agents and Environments, Good Behavior: the concept of Rationality, The Nature of Environments, The structure of Agents, How the components of agent programs work.

Unit-2 (Problems Solving, Search and Control Strategies)

Solving Problems by Searching, Study and analysis of various searching algorithms. Implementation of Depth-first search, Problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies: Breadth-first search, Uniform-cost search, Depth-first search, Depth-limited search, Iterative deepening depth-first search, Bi-directional search Informed (Heuristic) Search Strategies: Greedy best-first search A* search: Minimizing the total estimated solution cost, Conditions for optimality: Admissibility and consistency, Optimality of A*, Memory-bounded heuristic search, Heuristic Functions, Generating admissible heuristics from sub problems: Pattern databases, Learning heuristics from experience. Beyond Classical Search: Local Search Algorithms and Optimization Problems: Hillclimbing search Simulated annealing, Local beam search, Genetic algorithms, Local Search in Continuous Spaces, Searching with Non-deterministic Actions: AND-OR search trees, Searching with Partial Observations. Adversarial Search and Constraint Satisfaction Problems, Study of min-max algorithm Adversarial Search: Games, Optimal Decisions in Games, The mini-max algorithm, Optimal decisions in multiplayer games, Alpha--Beta Pruning, Move ordering , Imperfect Real-Time Decisions, Evaluation functions, Cutting off search, Forward pruning, Search versus lookup, Stochastic Games, Evaluation functions for games of chance, Partially Observable Games Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems, Variations on the CSP formalism, Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs, Local Search for CSPs, Alpha-beta pruning and CSP, Implementation aspects of minimax algorithm and CSP.

Unit- 3 (Knowledge Representations Issues, Predicate Logic, Rules)

Knowledge representation, KR using predicate logic, KR using rules. Reasoning System - Symbolic, Statistical: Reasoning, Symbolic reasoning, Statistical reasoning.

Unit-4 (Quantifying Uncertainty, Learning Systems)

Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Bayes' Rule and Its Use, Representing Knowledge in an Uncertain Domain, Other Approaches to Uncertain Reasoning, Rule-based methods for uncertain reasoning, representing vagueness: Fuzzy sets and fuzzy logic, Study of fuzzy logic and Decision trees, Implementation aspects of Decision trees. Learning from Examples: Forms of Learning, Supervised Learning, Learning Decision Trees, The decision tree representation, Expressiveness of decision trees, inducing decision trees from examples.

Unit-5 (Expert Systems)

Introduction, Knowledge acquisition, Knowledge base, working memory, Inference engine, Expert system shells, Explanation, Application of expert systems. Fundamentals of Neural Networks: Introduction and research history, Model of artificial neuron, Characteristics of neural networks, learning methods in neural networks, Singlelayer neural network system, Applications of neural networks. Fundamentals of Genetic Algorithms: Introduction, Encoding, Operators of genetic algorithm, Basic genetic algorithm.

Text Books

1. Rich, Elaine Knight, Kevin, Artificial Intelligence, Tata McGraw Hill.
2. Luger, George F, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education.

References

1. Nilsson, Nils J, Artificial Intelligence, Morgan Kaufmann.
2. Russell, Stuart J. Norvig, Peter, AI: A Modern Approach, Pearson Education.

TITLE OF COURSE: ARTIFICIAL INTELLIGENCE LAB

COURSE CODE: BCA691

L-T-P: 0-0-3

CREDITS: 3

SYLLABUS

Objectives:

In this course we will implement the basic components of an intelligent system, their functions, mechanisms, policies and techniques used in their implementation and examples.

Course Outcomes (CO):

The students will have a detailed knowledge of the concepts of artificial intelligence. Various applications of AI in different fields, aware of a variety of approaches to AI techniques.

Course Contents:

Unit-1: Introduction to AI and intelligent agents

Unit-2: Problem solving, Problem spaces and blind search techniques, informed search techniques, Constraint satisfaction problems

Unit-3: Knowledge representation and reasoning techniques, Logic programming, Logical agents, Game playing, planning,

Unit-4: Learning, Reasoning under uncertain situations,

Unit-5: Expert systems, Decision support systems, Domain specific AI applications.

List of AI Problems for Lab-

Problem 1: Solve “Tower of Hanoi” with only 2 disks.

Problem 2: Solve “Tower of Hanoi” with only 3 disks.

Problem 3: Solve “4-Queens” puzzle.

Problem 4: Solve “8-Queens” puzzle.

Problem 5: Solve “4-color map” problem.

Problem 6: Solve “8 – puzzle” take any initial and goal state.

Problem 7: Calculate the sum of n elements in an integer array. Also calculate its Polynomial function the determine its complexity using “*Big-O*”.

Problem 8: Find out the largest element in an square 2-D array. Also determine the “Big- O” of the algorithm. [Take size greater the 2x2]

Problem 9: Solve “Latin Square” problem.

Problem 10: Solve “Sudoku Problem” use any initial positions.

Problem 11: Solve “15-puzzle” problem using any initial and goal state.

Problem 12: Solve “Sudoku Problem” use any initial positions.

Problem 13: Code the following games software: Checkers, Chess.

Problem 14: Code the following games using software: Othello, Backgammon.

Problem 15: Code the following games using software: Bridge, Go.

Problem 16: Code the following games using software: Hex, 6x7.

Problem 17: Code the following games using software: Tetris, Tick-Tack-Toe.

Problem 18: Code the following games using software: rubik-cube, same game, mines.

Problem 19: Code the following games using software: Matches, Mines.

Text Books

1. Rich, Elaine Knight, Kevin, Artificial Intelligence, Tata McGraw Hill.
2. Luger, George F, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education.

References

1. Nilsson, Nils J, Artificial Intelligence, Morgan Kaufmann
2. Russell, Stuart J. Norvig, Peter, Artificial Intelligence: A Modern Approach, Pearson Education
3. Pearson Education
4. Negnevitsky, Michael, Artificial Intelligence: A Guide to Intelligent Systems, Addison-Wesley.

TITLE OF COURSE: CLOUD COMPUTING

COURSE CODE: BCA602A

L-T-P: 3-1-0

CREDITS: 4

SYLLABUS

Objectives:

The course covers the fundamental concepts and practical aspects of Service Oriented Architecture. The current software development and delivery model is service oriented in nature. The applications are inherently getting distributed and shared by multiple clients. Thus, there is a need to get an insight into service oriented architectures.

Course Outcomes (CO):

After having undergone the course, the student shall be able to understand the issues related with detailed design aspects and standards of SOA.

Course Contents:

Unit-1: SOA Fundamentals, Technologies, Benefits, Challenges and basic mechanisms associated with other computing service (Delivery models - SAS, IAS & PAS, Common Cloud deployment models and cloud characters), Security threats and mechanisms.

Unit-2: Introduction and fundamental of SOA, Benefits and Goals, SOA Manifesto, SOA and network management architecture, Service as web services, Discovery and publishing of web services, Service roles, Service models, Description of services with WSDL, Messaging with SOAP.

Unit-3: Exchange patterns of message, Service activity, Coordination, Composition, Types, Activation and registration process, Business activities, Orchestration, Composition of heterogeneous web services Choreography, Addressing, Reliable messaging, Correlation, Policies, Notification and eventing.

Unit-4: Security threats and mechanisms, Essential techniques, Patterns, Security architecture for service oriented solutions, Infrastructure, Middleware, Multitenancy concepts.

Text Books

1. Service Oriented Architecture, Concepts Technology and Design, Thomas Erl, Pearson Education, 2008
2. SOA in Practice: The Art of Distributed System Design, Nicolai M. Josuttis, O'Reilly, 2007

TITLE OF COURSE: DISTRIBUTED SYSTEM

COURSE CODE: BCA602B

L-T-P: 3-1-0

CREDITS: 4

SYLLABUS

Pre-requisite: Basic concepts in mathematics and digital signal processing.

Introduction:

1. Understand foundations of Distributed Systems.
2. Introduce the idea of peer to peer services and file system.
3. Understand in detail the system level and support required for distributed system.
4. Understand the issues involved in studying process and resource management.

Course Outcomes (CO):

At the end of the course, the student should be able to:

1. Discuss trends in Distributed Systems.

2. Apply network virtualization.
3. Apply remote method invocation and objects.
4. Design process and resource management systems

Course Contents:

Module 1: INTRODUCTION

Examples of Distributed Systems–Trends in Distributed Systems – Focus on resource sharing – Challenges. Case study: World Wide Web.

Module2: COMMUNICATION IN DISTRIBUTED SYSTEM

System Model – Inter process Communication – the API for internet protocols – External data representation and Multicast communication. Network virtualization: Overlay networks. Case study: MPI Remote Method Invocation And Objects: Remote Invocation – Introduction – Request-reply protocols – Remote procedure call – Remote method invocation. Case study: Java RMI – Group communication – Publish-subscribe systems – Message queues – Shared memory approaches – Distributed objects – Case study: Enterprise Java Beans -from objects to components.

Module3: PEER TO PEER SERVICES AND FILE SYSTEM

Peer-to-peer Systems – Introduction – Napster and its legacy – Peer-to-peer – Middleware – Routing overlays. Overlay case studies: Pastry, Tapestry- Distributed File Systems –Introduction – File service architecture – Andrew File system. File System: Features-File model -File accessing models – File sharing semantics Naming: Identifiers, Addresses, Name Resolution – Name Space Implementation – Name Caches – LDAP.

Module 4: SYNCHRONIZATION AND REPLICATION

Introduction – Clocks, events and process states – Synchronizing physical clocks- Logical time and logical clocks – Global states – Coordination and Agreement – Introduction – Distributed mutual exclusion – Elections – Transactions and Concurrency Control– Transactions -Nested transactions – Locks – Optimistic concurrency control – Timestamp ordering – Atomic Commit protocols -Distributed deadlocks – Replication – Case study – Coda.

Module5: PROCESS & RESOURCE MANAGEMENT Process Management: Process Migration: Features, Mechanism – Threads: Models, Issues, Implementation. Resource Management: Introduction- Features of Scheduling Algorithms –Task Assignment Approach – Load Balancing Approach – Load Sharing Approach.

TEXT BOOK:

1. George Coulouris, Jean Dollimore and Tim Kindberg, “Distributed Systems Concepts and Design”, Fifth Edition, Pearson Education, 2012.

REFERENCES:

1. Pradeep K Sinha, “Distributed Operating Systems: Concepts and Design”, Prentice Hall of India, 2007.
2. Tanenbaum A.S., Van Steen M., “Distributed Systems: Principles and Paradigms”, Pearson Education, 2007.
3. Liu M.L., “Distributed Computing, Principles and Applications”, Pearson Education, 2004.

4. Nancy A Lynch, “Distributed Algorithms”, Morgan Kaufman Publishers, USA, 2003.

TITLE OF COURSE: SOFTWARE PROJECT MANAGEMENT

COURSE CODE: BCA602C

L-T-P: 3-1-0

CREDITS: 4

SYLLABUS

Pre-requisite: Software Engineering, Software Project Management.

Introduction:

This course is strongly suggested for any student planning to take an internship in Computer Science. After an overview of the phases of the software lifecycle, current methodologies, tools, and techniques being applied to each phase will be discussed in depth with localized exercises given to reinforce learning of concepts.

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of software development and maintenance. To reach this goal, the following objectives need to be met:

1. Describe software process maturity framework.
2. Explain conventional software management and software economics.
3. Discuss software projects and project planning.
4. Analyze project tracking and control.
5. Assess the role of project closure analysis.
6. Successful Projects - What they have in common
7. Teams - Effective management and how to be an effective member
8. Risks - What are they and how they can affect a project
9. Plan-Driven process and methodologies, including classic and agile methodologies
10. Process - Choosing and adapting to the project
11. Estimation and Scheduling - How to get it right, and what can happen if you don't
12. Beyond the initial project - Product evolution, process improvement, and quality frameworks

Course Contents:

Module-1: PROJECT EVALUATION AND PROJECT PLANNING: Importance of Software Project Management – Activities Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation – Strategic program Management – Stepwise Project Planning.

Module-2: PROJECT LIFE CYCLE AND EFFORT ESTIMATION:Software process and Process Models – Choice of Process models – mental delivery – Rapid Application development – Agile methods – Extreme Programming – SCRUM – Managing interactive processes – Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points – COCOMO II A Parametric Productivity Model – Staffing Pattern.

Module-3: ACTIVITY PLANNING AND RISK MANAGEMENT: Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Forward Pass & Backward Pass techniques – Critical path (CRM) method – Risk identification – Assessment – Monitoring – PERT technique – Monte Carlo simulation – Resource Allocation – Creation of critical patterns – Cost schedules.

Module-4: PROJECT MANAGEMENT AND CONTROL:Framework for Management and control – Collection of data Project termination – Visualizing progress – Cost monitoring – Earned Value Analysis- Project tracking – Change control- Software Configuration Management – Managing contracts – Contract Management.

Module-5: STAFFING IN SOFTWARE PROJECTS:Managing people – Organizational behavior – Best methods of staff selection – Motivation – The Oldham-Hackman job characteristic model – Ethical and Programmed concerns – Working in teams – Decision making – Team structures – Virtual teams – Communications genres – Communication plans.

Text Books

1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Fifth Edition, Tata McGraw Hill, New Delhi, 2012.

Reference Books

1. Robert K. Wysocki “Effective Software Project Management” – Wiley Publication,2011.
2. Walker Royce: “Software Project Management”- Addison-Wesley, 1998.
3. Gopalaswamy Ramesh, “Managing Global Software Projects” – McGraw Hill Education (India), Fourteenth Reprint 2013.

TITLE OF COURSE: REAL TIME OPERATING SYSTEM

COURSE CODE: BCA602D

L-T-P: 3-1-0

CREDITS: 4

SYLLABUS

Introduction:

In several software applications, especially in embedded application, the operating system is required to support the application to meet the timing constraints. The operating system achieves this by deploying suitable scheduling algorithms. A major problem arises, when the real-time tasks share resources. Priority inversions can take place in this case, unless suitable techniques are deployed. Starting with a brief introduction to real-time operating systems, we first discuss the important real-time task/thread scheduling algorithms and resource sharing protocols. An effort towards standardization of real-time operating systems has come to be known as POSIX-RT. We review POSIX-RT requirements. Besides, we review several commercial and open source real-time operating systems.

Course Outcomes (CO):

Syllabus deals with issues in real time operating systems, importance of deadlines and concept of task scheduling. Student will be able to understand and design real time operating systems which are backbone of embedded industry.

1. Student will be able to summarize the issues in real time computing
2. Student will be able to explain and give examples of real time operating systems.
3. Student will be able to solve scheduling problems and can apply them in real time applications in industry.
4. Student can also design an RTOS and will be able to interpret the feasibility of a task set to accomplish or not.
5. Analyze the situation of fault occurrence and will be able to apply solutions accordingly

Module I: Introduction to Real time systems

Issues in real time computing, Structure of real time system, Need for RTOS, Task classes, Performance measures for real time system: Properties, traditional performance measures, perform ability, cost functions and hard deadlines, and Estimating program run times. Introduction LINUX/ UNIX OS

Module II: Embedded software and Task Scheduling

Examples of embedded system their characteristics and their typical hardware components, embedded software architectures, Scheduling algorithms: round robin, round robin with

interrupts, function queue scheduling, real time operating system selection, CPU scheduling algorithms: Rate monotonic, EDF, MLF. Priority Scheduling, Priority Ceiling and Priority inheritance, Real time operating system: Tasks and task states, shared data and reentrancy semaphores and shared data, use of semaphores, protecting shared data

Module III: Features of Real Time Operating System

Messages. Queues, mailboxes, pipes, timer function, events, memory management. Interrupt basic system design using an RT (OS design principles, interrupt routines, task structures and priority.), Current research in RTOS, Case Studies: Vx Works and Micro OS-II

Module IV: Real Time Databases

Real time v/s general purpose databases, main memory databases, transaction priorities transaction aborts, concurrency control issues: pessimistic concurrency control and optimistic concurrency control, Disk scheduling algorithms.

Module V: Fault Tolerance Techniques

Causes of failure, Fault types, Fault detection, Fault and error containment, Redundancy: hardware redundancy, software redundancy, Time redundancy, information redundancy, Data diversity, Integrated failure handling.

Text Books:

1. An Embedded Software Primer, David E. Simon Pearson Education Asia Publication ISBN-13: 9780201615692
2. Real Time Systems, C.M. Krishna and Kang G. Shin, TMH Publication ISBN 13: 9780070701151

Reference Books:

Embedded system: Architecture Programming and Design, Raj kamal, TMH Publication ISBN 13: 9780070667648

TITLE OF COURSE: ANDROID PROGRAMMING

COURSE CODE: BCA693A

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Introduction:

Android is an open source and Linux-based Operating System for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led

by Google, and other companies. Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android. The extendable application environment of Android is specifically designed to suit industry requirements to tap the mobile CONSUMER MARKET and offers immense scope for Android application developers. Greater part of Android application development is carried out in Java and this Android Course focuses on Java based application development.

Course Outcomes (CO):

The purpose of this course is to teach you how to program using Java and apply this knowledge to Android Platform in smart phones. This course will give you an opportunity to learn the currently hottest technology and to use the skills you have gained as an undergraduate student to make marketable software in smart phones. We will learn the general theme of software engineering for all topics. All topics will address fundamental Java programming language faced by professionals in IT industry. Another objective of this course is to provide students an environment to experience the life-cycle of software development. We will have projects to make marketable software for smart phone by combine Android and Java together. Here are pictures of the idea:

Upon successful completion of this course students will be able to ...

1. Understand the main concepts of mobile application development.
2. Understand the basic components of Android as an open source software.
3. Understand concepts of Intents, Broadcast receivers and working in the background
4. Compare between Android Views, Activities, and Fragments.
5. Compare between Intents, Broadcast receivers, and services
6. Compare between the basic Android layouts.
7. Be familiar with Android Studio as an IDE for developing Android applications
8. Be familiar with the errors that could be encountered while implementing an application
9. Use the Android SDK to implement mobile applications and run them on mobile devices.
10. Work in groups as well as individually to design and implement mobile applications
11. Work individually to handle errors while implementing Android applications

Course Contents:

Module-1: Introduction to Android

Introduction to Android Platform, Android Stack, Android Versions and Installing Android SDK and updating SDK components, Eclipse, IDEs and ADT plug-in, Using the Emulator, Android vs. Other mobile platforms

Module-2: Your First Android Application

Application Life Cycle, Application Components, Activity life cycle, Manifest File, Layout XML Code, Strings, The R File, Java Source Code, Java based layout vs. xml based layout, Eclipse Visual Layout Editor, Logging

Module-3: UI Design for Android

Using different layouts – LinearLayout, TableLayout etc, Drawable Resources, Resolution and density independence, Working with common widgets, Working with ListView and Adapters, Creating and using option menu, Working with preferences, Working with Dialogs and Toasts, Working with Graphics and Animation

Module-4: Introducing Intents

Intents, Intent filters, Invoking activities by class name and URI, Sharing data using Extras Bundle and URI parameters, working with Tabs and Fragments

Module-5: Files and Database

Using File System, Introducing SQLite on Android, Database Connectivity, Cursors and content values, Using Content Provider to share data, Understanding Security model

Module-6: Working in background

Introducing Service and its life cycle, Creating and starting a service, Types of services, Working with multi-threading and AsyncTask, Broadcast receivers, Triggering receivers with intents, Responding to system events using Broadcast receivers, Using Alarm

Module-7: Using System Services and Web Services

Using Location based Services, Telephony and SMS services, Bluetooth, Network and WiFi. Multimedia and Camera, Accessing Internet and Web Services from Android Applications

Module-8: Project

Understanding the requirement, Designing the interface and architecture, Best practices regarding application design and development, Writing code and testing it, Preparing application for Publishing, Publishing to Android Market and Physical device

Text Books:

1. Programming Android: Java Programming for the New Generation of Mobile Devices Second Edition by ZigurdMednieks, Laird Dornin, G. Blake Meike, Masumi Nakamura, O'REILLY
2. Head First Android Development: A Brain-Friendly Guide 2nd Edition by David Griffiths, O'REILLY

Reference Books:

1. Professional Android, 4th Edition, By Reto Meier, Ian Lake,

TITLE OF COURSE: ADMINISTRATION OF SQL SERVER

COURSE CODE: BCA693B

L-T-P: 0-0-3

CREDITS: 1.5

SYLLABUS

Pre-requisite: Knowledge is also assumed of basic concepts in data structures, programming languages.

Introduction:

The role of a data scientist is to turn raw data into actionable insights. Much of the world's raw data—from electronic medical records to customer transaction histories—lives in organized collections of tables called relational databases. Therefore, to be an effective data scientist, you must know how to wrangle and extract data from these databases using a language called SQL (pronounced ess-que-ell, or sequel)

Course Outcomes (CO):

This course will serve to broaden the student's understanding of the issues and latest developments in the area of SQL Server 2012. To reach this goal, the following objectives need to be met:

1. Features of DBMS & Models for designing databases.
2. Describe the nuances of Data retrieval methods.
3. Apply normalization techniques in DB design.
4. Perform concurrency and Transaction Management operations

Course Contents:

Module-1: Configuration of SQL Server 2012

Evaluating Installation Requirement of SQL 2012, Designing the installation, Planning scale up v/s scale out basics of SQL 2012, Shrinking and growing Database, Designing storage of new database, Capacity constraints, Standby database for reporting purpose, Window level and service level security for SQL 2012, Performing core model installation of SQL 2012, SQL Server Edition, Capacity, Licensing, Installing and Upgrading from SQL Server 2005, 2008 R2 to SQL 2012

Module-2: Roles of DBA & Database Developers

Production DBA, Development DBA, Architect DBA, ETL DBA, OLAP DBA, Basic Duties of DBA, Basic Knowledge of DBCC Commands

Module-3: I/O Planning & RAID & Window (2012) Cluster Configuration

I/O Fundamental, Window Cluster Configuration, Active & Passive Cluster Configuration, DNS & IP Benchmarking, Network Level Security and Antivirus, RAID Fundamental, Independent RAID, SAN Advantages

Module-4: Creating DB & DB Snapshot

Understanding Database, Create User Database, Viewing Database Details along with different kind of Graphs(2012) and customize level reporting, Fundamental of Database Snapshot, Managing Full Text Indexing, Configuring File stream, Configure File Table

Module-5: Fundamentals of Indexes

What is Index?, Understanding Anatomy of Balanced Tree (B-Tree), Understanding Index Type and Structure, Tracking missing indexes, Types of Index, Index Architecture, Index Maintenance & Tuning, Indexed Views

Module-6: Backup Fundamentals

Backup Fundamentals, Requirement of Backup, Backing up Replicated Database, System database and Mirrored database, Types of Backup, Advantages of Transactional log backup, Recovery Models & Logging Information, Backup of System Database

Module-7: Fundamentals of Restore

What is Restore?, Performing File Restores, Performing Page Restore, Restoring database protected with Transparent data encryption, Restore & Recovery Concepts, Restoring Database from Backup or Database

Module-8: Transaction & Locking Mechanism

What is Transaction?, ACID Properties, Isolation Levels, Types of New Isolation level in SQL Server 2012, Fundamentals of Locks, Row, Page & Table level Locks, Advantage and Disadvantage of Lock in OLTP Systems

Module-9: High Availability of Server

Disaster Recover, Failover Clustering, Log Shipping, Database Mirroring, What is Replication?, Types of Replication, Clustering Always on Feature of SQL 2012, Multi Sub net Failover Clustering

Text Books

1. SQL Server 2012 Administration, Microsoft Press
2. SQL Server 2012 Unleashed, BPB Publication

References

1. Amazing Power of SQL Server 2012, PHI Publication
2. Administration of SQL Server 2012 by Bharat & Company

TITLE OF COURSE: ESP & SDP - VI

COURSE CODE: GSC601

L-T-P: 0-1-0

CREDITS: 1

Introduction:

The course is on GATE exam preparation, revision and advanced problems in quantitative aptitude, reasoning, Verbal English etc.

Course contains:

Module-1

Stream wise GATE & IES syllabus.

SKILL DEVELOPMENT FOR PROFESSIONALS IV

Module-1

Revision and Advanced Problems in Quantitative Aptitude

- 1) Numbers (+, -, x, etc), Percentages, Ratio, Partnership, Linear Equations, Profit & Loss
- 2) Averages, Mixtures & Allegations, Number System, Time and Work
- 3) Simple & Compound Interest, Other / Misc Quantitative Apt., Indices and Surds, Quadratic Equations
- 4) Permutations & Combinations, Probability, Geometry, Mensuration
- 5) Data Interpretation, Various Charts, Diagrams, Tables

Module-2

Revision and Advanced Problems in Reasoning

- 1) Coding, Series & Numbers, Blood Relations, Analogy
- 2) Cubes, Data Sufficiency, Non-Verbal Reasoning
- 3) Syllogisms, Puzzles, Machine I/O, Inequality
- 4) Seating Arrangement, Calendar / Clock
- 5) Statements, Other / Misc Logical Reasoning, Decision Making (Ethics)

Module-3

Revision and Advanced Questions in Verbal English

- 1) Grammar,
- 2) Clauses,
- 3) Spotting errors,
- 4) Sentence Correction,
- 5) Blanks,
- 6) Reading Comprehensions,
- 7) Vocabulary

Newspaper reading: The Hindu & Economic Times.