SRI DHARMASTHALA MANJUNATHESHWARA COLLEGE (AUTONOMOUS)
UJIRE – 574 240
DAKSHINA KANNADA, KARNATAKA STATE
(Re-Accredited by NAAC at 'A++' Grade)
Ph: 08256-236221, 236101(O), Fax: 236220
e-Mail:sdmcollege@rediffmail.com, sdmcollege@sdmcujire.in, Website: www.sdmcujire.in

DEPARTMENTOF SOFTWARE & APP. DEVELOPMENT

Syllabus of

BACHELOR OF VOCATIONAL

[B VOC] PROGRAMME

ON

SOFTWARE & APP. DEVELOPMENT

(SEMESTER SYSTEM)

2019- 2020 onwards.

BOS meeting held on 19-08-2023
Academic Council meeting held on 02-09-2023

Preamble:

UGC Introduction to B.Vocational Courses

It has been a long felt necessity to align higher education with the emerging needs of the economy to ensure that the graduates of higher education system have adequate knowledge and skills for employment and entrepreneurship. The higher education system has to incorporate the requirements of various industries in its curriculum, in an innovative and flexible manner while developing a holistic and well-groomed graduate. Ministry of HRD, Government of India had issued an Executive Order in September 2011 for National Vocational Education Qualification Framework (NVEQF). Subsequently, Ministry of Finance, in pursuance of the decision of Cabinet Committee on Skill Development in its meeting held on 19December 2013, has issued a notification for National Skills Qualifications Framework (NSQF), which supersedes NVEQF. Under the National Skills Development Corporation, many Sector Skill Councils representing respective industries have/are being established. One of the mandates of Sector Skill Councils is to develop National Occupational Standards (NOSs) for various job roles in their respective industries. It is important to embed the competencies required for specific job roles in the higher education system for creating employable graduates. The University Grants Commission (UGC) has launched a scheme on skills development based higher education as part of college/university education, leading to Bachelor of Vocation (B.Voc.) Degree with multiple exits such as Diploma/Advanced Diploma under the NSQF. The B.Voc. Programmeis focused on universities and colleges providing undergraduate studies, which would also incorporate specific job roles and their NOSs along with broad based general education. This would enable the graduates completing B.Voc. To make a meaningful participation in accelerating India's economy by gaining appropriate employment, becoming entrepreneurs and creating appropriate knowledge.

Faculty of Bachelor of Vocational Program: BVOC006

Program Specific Outcomes: Application and Software Development (F3)

PSO1: Design a basic web site using HTML and CSS to demonstrate responsive web design.

PSO2: Demonstrate skills using word processor and spreadsheet presentation and Implement Static or dynamic web pages with validation using JavaScript objects by applying different event handling mechanism

PSO3: Describe XML using the user defined tags, DTD, Namespaces and Schemas with simple programs and discuss the concepts of PHP with associated programs.

PSO4: Understand and identify the models, components of a computer along with its connections, operating system concepts along with internet operation.

PSO5: Exhibit skills that can meet local, national and global work standards and discuss different ways to access the database through the web using examples and various server based software using different technologies.

B.VOC Software and Application Development

Introduction to Software & App. Development

Application and Software development is the discipline of creating high-quality software systems in a systematic, controlled and efficient manner. It involves the application of engineering concepts, techniques, and methods to the design, development, deployment and maintenance of software systems. An application and software development program will develop professionals who have a mastery of principles, theory, practices, and processes necessary to produce quality software systems.

The Application and Software development education in SDM will focus on imparting the knowledge and training, which should enable students to harmonize theory with practice, concept with application, and problem with solution. It will prepare them to apply ably engineering principles, practices, and processes to design, develop, deploy, and maintain software systems. The program will lead to development of student's professional and interpersonal skills. It will help students to enhance their ability in oral and written communication, and their adaptability to team environments. The program will inculcate among students a strong sense of civic, professional and ethical responsibility. The program will also strive to develop a capacity for innovation and a passion for lifelong learning.

Bachelor of Vocational Course (BVOC) Degree Programme Pattern and Scheme of Examinations

I / II / III / IV Semesters

	Courses	No. of Courses	Instruction Hrs/week	Duration of Exam(hrs)	Marks			Credits
					IA	Exam	Total	
Group 1	5 Computer Application Courses	2T	2*3	2*3	2*20	2*80	2*100	2*2=4
		3P	3*6	3*3	3*30	3*120	3*150	3*3=9
Group 2	2 Languages	2T	2*2	2*3	2*20	2*80	2*100	2*2=4
Group 3	Elective Foundation	1T	1*2	1*2	1*10	1*40	1*50	2*1=2
					Semester Credit Total			**19**

V Semester

	Courses	No. of Courses	Instruction Hrs/week	Duration of Exam(hrs)	Marks			Credits
					IA	Exam	Total	
Group 1	7 Computer Application Courses	4T	4*3	4*3	4*20	4*80	4*100	4*2=8
		3P	3*6	3*3	3*30	3*120	3*150	3*3=9
					Semester Credit Total			**17**

VI Semester

	Courses	No. of Courses	Instruction Hrs/week	Duration of Exam(hrs)	Marks			Credits
					IA	Exam	Total	
Group 1	4 Computer Application Courses	4T	4*3	4*3	4*20	4*80	4*100	4*2=8
	Project work	Dissertation	20	3*3	90	Project Report: 260 & Presentation with VIVA: 100	450	9
					Semester Credit Total			**17**
							Total Credits	**110**

Bachelor of Vocational Course (BVOC)

Degree Programme Pattern and Scheme of Examinations

Semester - I

Group	Course Code	Course	Instruction Hrs/week	Duration of Exam(hrs)	Marks & Credits			
					IA	Exam	Total	Credits
1	BVOCASCT101	Introduction to Python Programming	3	3	20	80	100	2
	BVOCASCT102	Introduction to Internet Programming	3	3	20	80	100	2
	BVOCASP103	Introduction to Python Programming Laboratory	6	3	30	120	150	3
	BVOCASP104	Introduction to Internet Programming Laboratory	6	3	30	120	150	3
	BVOCASP105	Basic Computer Skills Laboratory	6	3	30	120	150	3
2	BVOCKAN101 / BVOCHIN101	Kannada / Hindi	3	3	20	80	100	2
	BVOCENG101	English	3	3	20	80	100	2
3	BVOCEF101	Elective Foundation	2	2	10	40	50	2
			32	**23**	**180**	**720**	**900**	**19**

Semester - II

Group	Course Code	Course	Instruction Hrs/week	Duration of Exam(hrs)	Marks & Credits			
					IA	Exam	Total	Credits
1	BVOCASCT151	Introduction To Web Programming Using Xml, Php	3	3	20	80	100	2
	BVOCASCT152	Database Management System	3	3	20	80	100	2
	BVOCASP153	Web Programming Laboratory	6	3	30	120	150	3
	BVOCASP154	Database Management System Laboratory	6	3	30	120	150	3
	BVOCASP155	Computer Programming Laboratory	6	3	30	120	150	3
2	BVOCKAN151 / BVOCHIN151	Kannada / Hindi	3	3	20	80	100	2
	BVOCENG151	English	3	3	20	80	100	2
3	BVOCEF151	Elective Foundation	2	2	10	40	50	2
			32	**23**	**180**	**720**	**900**	**19**

Semester - III

Group	Course Code	Course	Instruction Hrs/week	Duration of Exam(hrs)	Marks & Credits			
					IA	Exam	Total	Credits
1	BVOCASCT201	Python for Data Science	3	3	20	80	100	2
	BVOCASCT202	Object Oriented Programming with Java	3	3	20	80	100	2
	BVOCASP203	Object Oriented Programming With Java Laboratory	6	3	30	120	150	3
	BVOCASP204	Python For Data Science Laboratory	6	3	30	120	150	3
	BVOCASP205	Object Oriented Programming Laboratory	6	3	30	120	150	3
2	BVOCKAN201 / BVOCHIN201	Kannada / Hindi	3	3	20	80	100	2
	BVOCENG201	English	3	3	20	80	100	2
3	BVOCEF201	Elective Foundation	2	2	10	40	50	2
			32	**23**	**180**	**720**	**900**	**19**

Semester - IV

Group	Course Code	Course	Instruction Hrs/week	Duration of Exam(hrs)	Marks & Credits			
					IA	Exam	Total	Credits
1	BVOCASCT251	Mobile Application Development	3	3	20	80	100	2
	BVOCASCT252	Introduction To Go Programming	3	3	20	80	100	2
	BVOCASP253	Mobile Application Development Laboratory	6	3	30	120	150	3
	BVOCASP254	Introduction To Go Programming-Laboratory	6	3	30	120	150	3
	BVOCASP255	Data Structures Laboratory	6	3	30	120	150	3
2	BVOCKAN251 / BVOCHIN251	Kannada / Hindi	3	3	20	80	100	2
	BVOCENG251	English	3	3	20	80	100	2
3	BVOCEF251	Elective Foundation	2	2	10	40	50	2
			32	**23**	**180**	**720**	**900**	**19**

Semester - V

Group	Course Code	Course	Instruction Hrs/week	Duration of Exam(hrs)	Marks & Credits			
					IA	Exam	Total	Credits
1	BVOCASCT301	ASP .NET CORE Programming	3	3	20	80	100	2
	BVOCASCT302	Machine Learning With Python	3	3	20	80	100	2
	BVOCASCT303	Unix And Shell Programming	3	3	20	80	100	2
	BVOCASCT304	DevOps For Developers	3	3	20	80	100	2
	BVOCASP305	Machine Learning With Python-Laboratory	6	3	30	120	150	3
	BVOCASP306	Unix And Shell Programming-Laboratory	6	3	30	120	150	3
	BVOCASP307	DevOPs For Developers Laboratory	6	3	30	120	150	3
			30	**21**	**170**	**680**	**850**	**17**

Semester - VI

Group	Course Code	Course	Instruction Hrs/week	Duration of Exam(hrs)	Marks & Credits			
					IA	Exam	Total	Credits
1	BVOCASCT351	Internet Of Things	3	3	20	80	100	2
	BVOCASCT352	Computer Networks	3	3	20	80	100	2
	BVOCASCT353	Ethical Hacking Techniques	3	3	20	80	100	2
	BVOCASCT354	NoSQL Database	3	3	20	80	100	2
		Basic Mathematics (Optional Subject)						
	BVOCASP355	Project Work	20	3	90	360	450	9
			32	**15**	**170**	**680**	**850**	**17**

Common scheme of Practical Examination for I to VI Semesters

The practical examination in the concerned subject specified in the I Semester to VI Semester shall be conducted for 120 marks. There shall be two components – Problem solving and execution and Viva voce components. 120 marks can be distributed as follows.

Each Practical paper includes four Parts- PART A, PART B, PART C and PART-D. One question shall be asked in eachpart.

SL NO	DETAILS			MARKS	TOTAL
1	PART A	i	Problem solving and coding	10	25
		ii	Compiling the code and debugging	9	
		iii	Execution and testing	6	
2	PART B	i	Problem solving and coding	10	25
		ii	Compiling the code and debugging	9	
		iii	Execution and testing	6	
3	PART C	i	Problem solving and coding	10	25
		ii	Compiling the code and debugging	9	
		iii	Execution and testing	6	
4	PART D	i	Problem solving and coding	10	25
		ii	Compiling the code and debugging	9	
		iii	Execution and testing	6	
5	Record				10
6	Viva –Voce				10
				Total Marks	120

INTRODUCTION TO PYTHON PROGRAMMING SEMESTER – I			
Subject Code	**BVOCASCT101**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Learn Syntax and Semantics and create Functions in Python.
- Handle Strings and Files in Python.
- Understand Lists, Dictionaries and Regular expressions in Python.
- Implement Object Oriented Programming concepts in Python.

Course outcomes: The students should be able to:

CO1: Examine python syntax and semantics and be fluent in the use of python flow control andfunctions.

CO2: Demonstrate proficiency in handling strings and file systems.

CO3: Create, run and manipulate python programs using core data structures like lists, dictionaries anduse regular expressions.

CO4: Interpret the concepts of object-oriented programming as used in python

CO5: Implement exemplary applications related to network programming, web services and databasesin python

Module – 1	**Teaching Hours**
Introduction to Computers and Problem Solving Strategies	**9 Hours**

Introduction, History, Characteristics and Classification of Computers, Basic Applications, Components and Functions, Concept of Hardware and Software, Central Processing Unit (CPU): Basic Architecture, Input and Output Devices, Computer Memory, Classification of Computer Software, Representation of Data: Bits and Bytes, Problem Solving Strategies, Program Design Tools: Algorithms, Flowcharts, Pseudo codes, Types of Errors, Testing andDebugging.	
Module – 2	
Basics of Python Programming Features and History of Python, The Future of Python, Writing and Executing First Python Program, Literal Constants, Variables and Identifiers, Data Types, Input Operation, Comments, Reserved Words, Indentation, Operators and Expressions, Operation on Strings,Other Data Types. **Decision Control Statements** Introduction, Selection or Conditional Statements, Loops/ Iterative Statements, Nested Loops	**9 Hours**
Module – 3	
Functions and Modules Introduction, Function Definition, Function Call, Variable Scope and Life Time, Return Statement, Arguments, Recursive Functions, Modules, Packages, Standard Library Modules.	**9 Hours**

Module –4	
Strings: Concatenating, Appending and Multiplying, Built in String Functions, Slice Operations,Regular Expressions. **Data Structures**: Sequence, Lists, Tuples, Sets and Dictionary	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule
- The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Reema Thareja, Python Programming using Problem Solving Approach, OXFORD UniversityPress.
2. Allen B. Downey, **"Think Python: How to Think Like a Computer Scientist",** 2nd Edition,

   Green Tea Press, 2015.

Reference Books:

1. Joel Grus ,Data Science from Scratch First Principles with Python, O'Reilly Media,2016
2. T.R.Padmanabhan, Programming with Python, Springer Publications,2016

INTRODUCTION TO INTERNET PROGRAMMINGSEMESTER – I			
Subject Code	**BVOCASCT102**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Illustrate the Semantic Structure of HTML and CSS
- Compose forms and tables using HTML and CSS
- Examine JavaScript framework.

Course outcomes: The students should be able to:

CO1: Adapt html and css syntax and semantics to build web pages. CO2: Construct and visually format tables and forms using html and css

CO3: Develop client-side scripts using JavaScript and server-side scripts using php to generate anddisplay the contents dynamically.

CO4: Appraise the principles of object oriented development-using php

CO5: Inspect JavaScript frameworks like jQuery and backbone which facilitates developer to focus oncore features

Module – 1	**Teaching Hours**
Introduction to Fundamentals of Web Internet, WWW, Web Browsers and Web Servers; URLs; MIME; HTTP; Security; The web Programmers Toolbox, XHTML: Origin and Evolution of HTML and XHTML;	**9 Hours**

BasicSyntax; Standard XHTML Document Structure; Basic Text Markup.	
Module – 2	
Introduction to HTML: What is HTML and Where did it come from? HTML Syntax, Structure of HTML Documents, Quick Tour of HTML Elements, Semantics Mark-Up,HTML5 Semantic Structure Elements.	**9 Hours**

Module – 3	
HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Forms Control Elements, Tables and Forms Accessibility Micro formats, Advanced HTML Graphics-HTML canvas, HTML SVG, HTML media, HTML Video, HTML Audio, HTML YouTube.	**9 Hours**
Module –4	
CSS: What is CSS, CSS Syntax, Location of Styles, Selectors, The Cascade: How Styles Interact, The Box Model, CSS Text Styling, CSS Borders, CSS Margins, CSS Padding, CSS Height Width, CSS Outline, CSS Text, CSS Responsive Font Size, CSS Font Variant, CSS Links, Advanced Links Buttons, CSS lists, CSS Display, CSS Floats, CSS Navigation, CSSDrop Down, JavaScript: Client Side Scripting, What is JavaScript and What can it do?, JavaScript Design Principals, Where does Java Script Go, Syntax, Java Script Objects, The Document Object Model(DOM), Java Script Events, Forms, Java Script Statement, JS Syntax, JS Comments, JS Variables, JS Operators, JS Arithmetic, JS Data Types, JS Functions, JS Objects, JS Events, JS Strings, JS Events, JS String Methods, JS Array Methods, JS Array Sort, JS Dateget and set methods, JS Loop For, JS Loop While	**9 Hours**

Question paper pattern:
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
• PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules • In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule • The students will have to answer 4 full questions, selecting one full question from each module
Text Books:
1. Programming the World Wide Web-Robert W.Sebesta, 4th Edition, Pearson Education, 2008. Programming the World Wide Web, 7th edition, Robert W.Sebesta , Pearson Education, ISBN-9789332518827
Reference Books:
1. Internet and World Wide Web –How to program by Dietel and Nieto Pearson Education 2. The complete Reference Java 2 Third Edition by Patrick Naughton and Herbert Schildt. 3. Java Server Pages by Hans Bergstan.

INTRODUCTION TO PYTHON PROGRAMMING LABORATORY SEMESTER – I			
Laboratory Code	**BVOCASP103**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Basics of Python programming
- Decision Making and Functions in Python
- Install and run the Python interpreter
- Create and execute Python programs
- Understand the concepts of file I/O
- Be able to read data from a text file using Python

Course outcomes: The students should be able to:

CO1: Be skilled in the basics of python programming

CO2: Be skilled in decision making and functions in python

CO3: Install and run the python interpreter

CO4: Create and execute python programs

CO5: Understand the concepts of file i/o

CO6: Be able to read data from a text file using python

Laboratory Experiments:

1. A) Write a program to check whether the given year is leap year or not.

B) Write a program to determine whether the entered character is Vowel or not
2. A)Write a program to read a number and calculate the sum of its digits(E.g. Input number= 123 sum=6). B) Write a program to calculate GCD of two numbers.
3. Write a program to perform Addition, Subtraction, Multiplication, and Division and modulo operations on two integer numbers (Read the input from keyboard).
4. Write a program to read three integer numbers from keyboard and find the largest among three numbers.
5. Write a python program to find a key element using Binary Search algorithm
6. Develop a program to find a reverse of a positive integer and check for palindrome or not. Display appropriate message.
7. Write a program to count the number of digits, uppercase characters, lowercase characters and special characters in a given string.
8. Write a program to remove all duplicates from the list
9. Write a program that creates a list of 10 random integers. Then create ODD list and EVEN list that has all odd and even values in the list respectively.
10. Write a program that has dictionary of names of students and a list of their marks in four subjects. Create another dictionary from this dictionary that has name of the student and theirtotal marks. Find out topper and his/her score.
Conduction of Practical Examination: • All laboratory experiments are to be included for practical examination. • Students allowed picking four experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

INTRODUCTION TO INTERNET PROGRAMMING LABORATORY SEMESTER – I			
Laboratory Code	**BVOCASP104**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- To acquire knowledge and Skills for creation of Web Site considering both client- and server-side Programming.
- To create Web application using tools and techniques used in industry.
- To be well versed with XML and web services Technologies.
- To be familiarized with open source Frameworks for web development.

Laboratory Experiments:

1. Design a page having suitable background colour and text colour with title "My First Web Page" using all the attributes of the Font tag.
2. Create a HTML document giving details of your [Name, Age], [Address, Phone] and [Register Number, Class] aligned in proper order using alignment attributes of Paragraph tag.
3. Write HTML code to design a page containing some text in a paragraph by giving suitable heading style.
4. Create a page to show different character formatting (B, I, U, SUB, SUP) tags.
5. Write HTML code to create a Web Page that contains an Image at its centre
6. Create a web page with an appropriate image towards the left hand side of the page,

when userclicks on the image another web page should open.
7. Create web Pages using Anchor tag with its attributes for external links.
8. Create a web page for internal links; when the user clicks on different links on the web page it should go to the appropriate locations/sections in the same page.
9. Write a HTML code to create a web page with pink colour background and display moving message in red colour.
10. Create a web page, showing an ordered list of all First semester courses (Subjects).
11. Create a web page, showing an unordered list of names of all the B.Voc Programmers (Branches) in your institution.
12. Create a HTML document containing a nested list showing a content page of any book.
13. Create the following table in HTML with Dummy Data which contains Reg. Number, Student Name, Year/Semester and Date of Admission
14. Create a web page which divides the page in two equal frames and place the audio and video clips in frame-1 and frame-2 respectively
15. Create a web page which should generate following output which contains Frame-1, Frame-2 and frame-3
16. Create a web page using Embedded CSS and multimedia.
Course outcomes: The students should be able to:
CO1: Design a basic web site using html and css to demonstrate responsive web design. CO2: Implement static or dynamic web pages with validation using JavaScript objects by applying different event handling mechanism

Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking four experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

BASIC COMPUTER SKILLS LABORATORYSEMESTER – I			
Laboratory Code	**BVOCASP105**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Will learn and understand the Basics of Computers and apply the application tools like wordprocessor, spreadsheet and presentation.

Course outcomes: The students should be able to:

CO1: Understand and identify the models, components of a computer along with its connections,operating system concepts along with internet operation

CO2: Demonstrate skills using word processor

CO3: Demonstrate skills using spreadsheet presentation

CO4: Demonstrate skills using presentation

Laboratory Experiments:

Lab 1: Computer hardware and software: Understand and identify the models, components of a computer along with its connections, operating system concepts along with internet operation.

Identify and understand the models of Computers, Identify and understand front panel switches and back panel connections of a Computer system, Identify and understand the physical components of a Computer, Conduct computer system connection and understand the booting process, Study and Practice of Basic DOS Commands, Familiarization of GUI based Operating System Environment, Practice creating Icons

and Folders, Creating/Opening of file, Editing and saving the document, Copy, Cut and Paste operations, built-in utilities of OS like – Text Editors, Paint, Calculator etc. ,Practice browsing of different sites using Search Engine, Practice Creating E-Mail accounts, Sending, Receiving of E-Mails.

1. Identify and understand the models of Computers, Identify and understand front panel switches and back panel connections of a Computer system, Identify and understand the physicalcomponents of a Computer.
2. Conduct computer system connection and understand the booting process.
3. Study and Practice of Basic DOS Commands.
4. Familiarization of GUI based Operating System Environment.
5. Practice creating Icons and Folders, Creating/Opening of file, Editing and saving the document,Copy, Cut and Paste operations, built-in utilities of OS like – Text Editors, Paint, Calculator, etc.
6. Practice browsing of different sites using Search Engine.

   Practice Creating E-Mail accounts, Sending, Receiving of E-Mails

Lab 2: Demonstrate skills using word processor

Create a Business Letter and Personal Letter, Create a Company Letter head, and Create a Simple Newsletter with minimum three columns. Insert a Clip Art in the Newsletter, Create a Resume for a Job Application, Create the Cover Page of a Project Report (use Word Art, insert Picture Image), Prepare the class time table of your class.

1. Create a Business Letter and Personal Letter.
2. Create a Company Letter head.
3. Create a Simple Newsletter with minimum three columns. Insert a Clip Art in the Newsletter.
4. Create a Resume for a Job Application.
5. Create the Cover Page of a Project Report (use Word Art, insert Picture Image). Prepare the classtime table of your class.

Lab 3: Demonstrate skills using spreadsheet presentation

1. Create a worksheet with five columns. Enter ten records and find the sum of all columns using auto sum feature.
2. You have a monthly income of Rs.11000. Your monthly expenditures are: Rent- Rs 3500, Food- Rs. 1500, Electricity- Rs.110, Phone- Rs. 160, and Cable TV-Rs. 300. Prepare a worksheet with the Monthly Income, the Monthly Expenditures listed and summed, monthly savings amount (what's left over each month) calculated, and the amount saved per day (assuming 30 days in a month).
3. Create a worksheet containing the Pay details (containing Basic pay, DA, HRA ,Other Allowance Deductions- PF, PT, Insurance, Gross and Net salary) of the Employees using Formulas.
4. Create a Simple Bar Chart to highlight the sales of a company for three different periods.
5. Create a Pie Chart for a sample data and give legends.

Lab 4: Demonstrate skills using presentation

1. Using presentation tool, Create a simple Presentation consisting of 4-5 slides about Input andOutput Devices.
2. Create a presentation about a Book containing Title, Author, Publisher and Contents.
3. Create an automated (with timings & animation) Presentation with five slides about differentModels of Computers. Use Presentation tool.

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- Students allowed picking four experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

Change of experiment allowed only once and marks allotted to the procedure part to made zero.

References:
• Computer Fundamentals Concepts, Systems, Application, D.P.Nagapal, S.Chand Publication,RP-2014, ISBN: 81-219-2388-3 • http://www.tutorialsforopenoffice.org/ • http://www.libreoffice.org/get-help/documentation/
Software Tools:
• Any open source tool or equivalent proprietary tools.

INTRODUCTION TO WEB PROGRAMMING USING XML, PHP SEMESTER – II			
Subject Code	**BVOCASCT151**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- To study the concepts of web application development such as XHTML, XML, PHP, Java websoftware, and Database access through JDBC and PHP.

Course outcomes: The students should be able to:

CO1: Discuss the fundamentals of web and concept of xhtml.

CO2: Describe different concepts of JavaScript and xhtml documents and construct dynamic documentswith JavaScript.

CO3: Describe xml using the user defined tags, dtd, namespaces and schemas with simple programsCO4: Discuss the concepts of php with associated programs

CO5: Discuss different ways to access the database through the web using examples. Discuss various server based software using different technologies

Module – 1	**Teaching Hours**
Introduction to XML: Introduction to XML, How Can XML be used?, XML Tree, XML Syntax Rules, XML Elements, XML Attributes, XML Namespaces, Displaying XML,XML Parser, XML DOM,XML and XQuery, XML and XPath, XML, XLink	**9 Hours**

and XPointer, XML Validator, XML DTD, XML Schema, XML on the Server, AJAX Introduction.	
Module – 2	
Introduction to PHP: Introduction to PHP, PHP Installation, PHP Syntax, PHP Comments, PHP Variables, PHP echo and, print Statements, PHP Data Types, PHP Strings, PHP Numbers, PHP Constants, PHP Operators, PHP if...else...else if Statements, PHP switch Statement, PHP Loops, PHP Functions, PHP Arrays, PHP Global Variables –Super global.	**9 Hours**
Module – 3	
PHP Forms And PHP Advanced: PHP Form Handling, PHP Form Validation, PHP Forms - Required Fields, PHP Forms - Validate E-mail and URL, PHP - Complete Form Example, PHP Advanced, PHP Include Files, PHP File Handling, PHP File Open/Read/Close, PHP File Create/Write, PHP File Upload, PHP Cookies, PHP Sessions, PHP Filters, PHP Filters Advanced.	**9 Hours**
Module –4	
MySQL Database PHP MySQL Database, PHP Connect to MySQL, PHP Create a MySQL Database, PHP MySQL Create Table, PHP MySQL Insert Data, PHP MySQL Get Last Inserted ID, PHP MySQL Insert Multiple Records, PHP MySQL Prepared Statements, PHP MySQL Select Data, PHP MySQL Use The WHERE Clause, PHP MySQL Use The ORDER BY Clause, PHP MySQL Update Data.	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule
- The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Programming the World Wide Web, 7th edition, Robert W.Sebesta , Pearson Education, ISBN-9789332518827

Reference Books:

1. http://www.tutorialspoint.com/
2. http://www.w3schools.com/
3. Web Programming – Building Internet Applications, 3rd edition, Chris Bates, Wiley publisher
4. Web Technologies— HTML,JavaScript,PHP,java,JSP,ASP.Net,XML & Ajax – Black Book,Wiley, ISBN : 978-81-7722-997-4
5. PHP A Begineer‘s Guide --- Vikram Vaswami , TMH publishers. ISBN: 13:978-007-014069-1

DATABASE MANAGEMENT SYSTEMSEMESTER – II			
Subject Code	**BVOCASCT152**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Provide a strong foundation in database concepts, technology, and practice.
- Practice SQL programming through a variety of database problems.
- Demonstrate the use of concurrency and transactions in database
- Design and build database applications for real world problems.

Course outcomes: The students should be able to:

CO1: Identify, analyze and define database objects, enforce integrity constraints on a database usingrdbms.

CO2: Use structured query language (sql) for database manipulation.

CO3: Design and build simple database systems develop application to interact with databases

Module – 1	**Teaching Hours**
Databases and Data Base Users: Introduction, An Example, Characteristics of the database approach, Actors on the scene, Workers behind the scene, Advantages of using the DBMS Approach, A Brief History of Database Applications, When Not to use a DBMS Database System Concepts and Architecture - Data Models, Schemas, and Instances, Three- Schema Architecture and Data Independence,	**9 Hours**

Database Languages and Interfaces, The Database System Environment, Centralized and Client /Server Architectures for DBMSs, Classification of database Management System. Functional Dependencies and normalization for relational databases: Informal Design guidelines for relation schemas, Functional dependencies, Normal forms based on primary keys, General Definition of second and third normal forms, Boyce-codd Normal form.	
Module – 2	
Data Modelling Using the Entity-Relationship(ER) Model: Using High-Level Conceptual Data Models for Database Design, An example Database Application, Entity Types, Entity Sets, attributes and keys, Relation Types, Relationship Sets, roles and structural constraints, Weak Entity Types, Refining the ER Design for the Company Database, ER Diagrams, naming, conventions and design issues, Relationship Types of Degree Higher Than Two.	**9 Hours**
Module – 3	
Relational Data Model and Relational Data Base Constraints: Relational Model concepts,Relational Model Constraints and relational database schemas, Update Operation, Transaction and Dealing with constraints violations.	**9 Hours**
Module –4	

Schema Definition, constraints, queries and views: SQL Data Definition and data types, Specifying constraints in SQL, Schema Change statement in SQL, Basic queries in SQL, More Complex SQL queries, INSERT, DELETE and UPDATE statements in SQL, Specifying constraints and Assertions and Triggers, Views(Virtual Tables) in SQL	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule
- The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Fundamentals of Database Systems, Sixth edition, 2014, Ramez Elmasri, Shamkan B. Navathe,Pearson Education, ISBN- 9788131792476
2. http://www.w3resource.com/mongodb/nosql.php --for NoSQL - UNIT VI

Reference Books:

1. Fundamentals of Database Management Systems, Mark L. Gillenson, 2009, Wiley India
2. Database Management Systems – Raghu Rama krishnan and Johannes Gehrke – 3rd Edition,McGraw-Hill, 2003
3. DBMS a practical approach, by E R Rajiv Chopra, S Chand publications.
4. http://elearning.vtu.ac.in/10CS54.html
5. http://www.tutorialspoint.com/dbms/
6. http://www.indiabix.com/technical/dbms-basics/
7. http://beginner-sql-tutorial.com/sql.htm

WEB PROGRAMMING LABORATORY SEMESTER – II			
Laboratory Code	**BVOCASP153**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- To study the concepts of web application development such as XHTML, XML, PHP, Java websoftware, and Database access through JDBC and PHP.

Course outcomes: The students should be able to:

CO1: Describe xml using the user-defined tags, dtd, namespaces and schemas with simple programs.

CO2: Discuss the concepts of php with associated programs

CO3: Discuss different ways to access the database through the web using examples. Discuss various server based software using different technologies

Laboratory Experiments:

1. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.

2. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to10and outputs HTML text that displays the resulting values in an HTML table format.

3. Write a JavaScript code that displays text “TEXT-GROWING” with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays “TEXT-SHRINKING” in BLUE color. Then the font size decreases to5pt

4. Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions forthe following problems: a) Parameter: A string b) Output: The position in the string of the left-most vowel c) Parameter: A number d) Output: The number with its digits in the reverse order
5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and email id. Make up sample data for 3students. Create a CSS style sheet and use it to display the document.
6. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
7. Write a PHP program to a) Implement simple operations. b) Find the transpose of a matrix. c) Multiplication of two matrices. d) Addition of two matrices
8. Write a PHP program named states.py that declares variable states with value “MississippiAlabama Texas Massachusetts Kansas". write a PHP program that does the following: a) Search for a word in variable states that ends in xas. Store this word in element0 of a list namedstates List. b) Search for a word in states that begins with k and ends in s. Perform a case-insensitive comparison. [Note: Passing re.Ias a second parameter to method compile performs a case-insensitive comparison.] Store this word in element1of

states List.

c) Search for a word in states that begins with M and ends in s. Store this word in element2 of thelist.

d) Search for a word in states that ends in a. Store this word in element 3 of the list

9. Write a PHP program to sort the student records which are stored in the database using selection sort.

10. Write a PHP program to display a digital clock which displays the current time of the server.

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- Students allowed picking four experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

Change of experiment allowed only once and marks allotted to the procedure part to made zero.

DATABASE MANAGEMENT SYSTEM LABORATORYSEMESTER – II			
Laboratory Code	**BVOCASP154**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Foundation knowledge in database concepts, technology and practice to groom students intowell-informed database application developers.
- Strong practice in SQL programming through a variety of database problems.
- Develop database applications using front-end tools and back-end DBMS

Course outcomes: The students should be able to:

CO1: Create, update and query on the database.

CO2: Demonstrate the working of different concepts of dbms

CO3: Implement, analyze and evaluate the project developed for an application

Laboratory Experiments:

PART-A: SQL Programming (Max. Exam Marks. 70)

1. Design, develop, and implement the specified queries for the following problems using Oracle,MySQL, MS SQL Server, or any other DBMS under LINUX/Windows environment.
2. Create Schema and insert at least 5 records for each table. Add appropriate databaseconstraints.

PART-B: Mini Project (Max. Exam Marks. 50)

1. Use Java, C#, PHP, Python, or any other similar front-end tool. All applications

must be demonstrated on desktop/laptop as a stand-alone or web based application (Mobile apps onAndroid/IOS are not permitted.)

Lab 1: Consider the following schema for a Library Database:

BOOK (Book_id, Title, Publisher_Name, Pub_Year) **BOOK_AUTHORS** (Book_id, Author_Name) **PUBLISHER** (Name, Address, Phone) **BOOK_COPIES** (Book_id, Branch_id, No-of_Copies)

BOOK_LENDING (Book_id, Branch_id, Card_No, Date_Out, Due_Date)

LIBRARY_BRANCH (Branch_id, Branch_Name, Address)

Write SQL queries to:

1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copiesin each branch, etc.
2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2019 toJun 2020
3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulationoperation.
4. Partition the BOOK table based on year of publication. Demonstrate its working with a simplequery.
5. Create a view of all books and its number of copies that are currently available in the Library.

Lab 2: Consider the following schema for Order Database:

SALESMAN (Salesman_id, Name, City, Commission)

CUSTOMER (Customer_id, Cust_Name, City, Grade, Salesman_id)

ORDERS (Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)

Write SQL queries to:

1. Count the customers with grades above Bangalore's average.
2. Find the name and numbers of all salesmen who had more than one customer.
3. List all salesmen and indicate those who have and don't have customers in their cities (UseUNION operation.)
4. Create a view that finds the salesman who has the customer with the highest order of a day.
5. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders mustalso be deleted.

Lab 3: Consider the schema for Movie Database:

ACTOR (Act_id, Act_Name, Act_Gender)

DIRECTOR (Dir_id, Dir_Name, Dir_Phone)

MOVIES (Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id)

MOVIE_CAST (Act_id, Mov_id, Role) RATING (Mov_id, Rev_Stars)

Write SQL queries to:

1. List the titles of all movies directed by 'Hitchcock'.
2. Find the movie names where one or more actors acted in two or more movies.
3. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOINoperation).
4. Find the title of movies and number of stars for each movie that has at least one rating

and findthe highest number of stars that movie received. Sort the result by movie title.

5. Update rating of all movies directed by 'Steven Spielberg' to 5.

Lab 4: Consider the schema for College Database:

STUDENT (USN, SName, Address, Phone, Gender)

SEMSEC (SSID, Sem, Sec)

CLASS (USN, SSID)

SUBJECT (Subcode, Title, Sem, Credits)

IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

Write SQL queries to:

1. List all the student details studying in fourth semester 'C' section.
2. Compute the total number of male and female students in each semester and in each section.
3. Create a view of Test1 marks of student USN '1BI15CS101' in all subjects.
4. Calculate the FinalIA (average of best two test marks) and update the corresponding table forall students.
5. Categorize students based on the following criterion: If FinalIA = 17 to 20 then CAT = 'Outstanding' If FinalIA = 12 to 16 then CAT = 'Average' If FinalIA< 12 then CAT = 'Weak'Give these details only for 8th semester A, B, and C section students.

Lab 5: Consider the schema for Company Database:

EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)**DEPARTMENT** (DNo, DName, MgrSSN, MgrStartDate) **DLOCATION** (DNo,DLoc)

PROJECT (PNo, PName, PLocation, DNo)

WORKS_ON (SSN, PNo, Hours)

Write SQL queries to:
1. Make a list of all project numbers for projects that involve an employee whose last name is‘Scott’, either as a worker or as a manager of the department that controls the project. 2. Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10percent raise. 3. Find the sum of the salaries of all employees of the ‘Accounts’ department, as well as themaximum salary, the minimum salary, and the average salary in this department 4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.
Conduction of Practical Examination:
1. All laboratory experiments from part A are to be included for practical examination. 2. Mini project valuated for 40 Marks and 10 Marks for record. 3. Report should be prepared in a standard format prescribed for project work. 4. Students are allowed to pick two experiment from part-A 5. Strictly follow the instructions as printed on the cover page of answer script. **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

COMPUTER PROGRAMMING LABORATORYSEMESTER – II			
Laboratory Code	**BVOCASP155**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Write flowcharts, algorithms and programs.
- Familiarize the processes of debugging and execution.
- Implement basics of C programming language.
- Illustrate solutions to the laboratory programs.

Course outcomes: The students should be able to:

CO1: Write algorithms, flowcharts and program for simple problems.CO2: Correct syntax and logical errors to execute a program.

CO3: Write iterative and wherever possible recursive programs

CO4: Demonstrate use of functions, arrays, strings and structures in problem solving

Laboratory Experiments:

1. Familiarization with programming environment, concept of naming the program files, storing, compilation, execution and debugging. Taking any simple C- code.
2. Develop a program to solve simple computational problems using arithmetic expressions anduse of each operator leading to simulation of a Commercial calculator. (No built-in math function)
3. Develop a program to compute the roots of a quadratic equation by accepting the coefficients. Print appropriate messages.

4. Develop a program to find the reverse of a positive integer and check for palindrome or not. Display appropriate messages.
5. An electricity board charges the following rates for the use of electricity: for the first 200 units80 paise per unit: for the next 100 units 90 paise per unit: beyond 300 units Rs 1 per unit. All users are charged a minimum of Rs. 100 as meter charge. If the total amount is more than Rs. 400, then an additional surcharge of 15% of total amount is charged. Write a program to readthe name of the user, number of units consumed and print out the charges.
6. Introduce 1D Array manipulation and implement Binary search.
7. Implement using functions to check whether the given number is prime and display appropriate messages. (No built-in math function)
8. Develop a program to introduce 2D Array manipulation and implement Matrix multiplication and ensure the rules of multiplication are checked.
9. Develop a Program to compute Sin(x) using Taylor series approximation. Compare your result with the built-in Library function. Print both the results with appropriate messages.
10. Write functions to implement string operations such as compare, concatenate, string length. Convince the parameter passing techniques.
11. Develop a program to sort the given set of N numbers using Bubble sort.
12. Develop a program to find the square root of a given number N and execute foe all possible inputs with appropriate messages. Note: Do not use library function sqrt (n).
13. Implement structures to read, write and compute average marks and the students scoring aboveand below the average marks for a class of N students.

14. Develop a program using pointers to compute the sum, mean and standard deviation of all elements stored in an array of n real numbers
15. Implement recursive functions for Binary to Decimal Conversion
Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking four experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

PYTHON FOR DATA SCIENCESEMESTER – III			
Subject Code	**BVOCASCT201**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Explore Python language fundamentals, including basic syntax, variables, and types
- Create and manipulate regular Python lists
- Use functions and import packages
- Build Numpy arrays, and perform interesting calculations
- Create and customize plots on real data
- Supercharge your scripts with control flow, and get to know the Pandas Data Frame

Module – 1	**Teaching Hours**
Classes and Objects Classes and objects, Inheritance, Exception Handling**.**	**9 Hours**
Module – 2	
Using Numpy Basics of Numpy-Computation on Numpy-Aggregations-Computation on Arrays Comparisons, Masks and Boolean Arrays-Fancy Indexing-Sorting Arrays-Structured Data:NumPy's Structured Array.	**9 Hours**
Module – 3	

Data Manipulation with Pandas Introduction to Pandas Objects-Data indexing and Selection-Operating on Data in Pandas-Handling Missing Data-Hierarchical Indexing - Combining Data Sets	**9 Hours**
Module –4	
Visualization and Matplotlib Basic functions of matplotlib-Simple Line Plot, Scatter Plot-Density and Contour Plots- Histograms, Binning's and Density-Customizing Plot Legends, Colour Bars-Three- Dimensional Plotting in Matplotlib.	**9 Hours**
Course outcomes: The students should be able to:	
CO1: Advanced concepts of python like writing python scripts.CO2: Sequence and file operations in python. CO3: Use libraries like pandas, NumPy, matplotlib, scikit, and master the concepts like python machine learning, scripts, and sequence	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) • PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules • In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule • The students will have to answer 4 full questions, selecting one full question from each module	
Text Books:	

1. Jake VanderPlas ,Python Data Science Handbook - Essential Tools for Working with Data,O'ReilyMedia,Inc, 2016
2. Zhang.Y ,An Introduction to Python and Computer Programming, Springer Publications,2016.
3. ReemaThareja, Python Programming using Problem Solving Approach, OXFORD University Press.

Reference Books:

1. Joel Grus ,Data Science from Scratch First Principles with Python, O'Reilly Media,2016
2. T.R.Padmanabhan, Programming with Python, Springer Publications,2016DBMS a practicalapproach , by E R Rajiv Chopra, S Chand publications.

OBJECT ORIENTED PROGRAMMING WITH JAVASEMESTER – III			
Subject Code	**BVOCASCT202**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Learn fundamental features of object oriented language and JAVA
- Set up Java JDK environment to create, debug and run simple Java programs.
- Learn object oriented concepts using programming examples.
- Study the concepts of importing of packages and exception handling mechanism.
- Discuss the String Handling examples with Object Oriented concepts.

Course outcomes: The students should be able to:

CO1: Understand how to install and use a good java development environment.

CO2: Use an integrated development environment to write, compile, run, and test simple object- oriented java programs

CO3: Demonstrate basic problem solving skills: analyzing problems, modeling a problem as a system of objects, creating algorithms, and implementing models and algorithms in an object- oriented computer language

CO4: Read and make elementary modifications to java programs that solve real-world problems

Module – 1	**Teaching Hours**
Java introduction, java syntax, java comments, java variables, java data types, java type	**9 Hours**

casting, java operators, java string, java match, java Booleans, java if else, java switch, javawhile loop, java for loop, java break/continue, java array.	
Module – 2	
Java methods, java method parameters, java method overloading, java classes, java oop, java classes and objects, java class attribute, java class methods, java constructors, java modifiers, java encapsulation, java packages/API, java inheritance, java polymorphism ,java inner classes, java abstraction.	**9 Hours**
Module – 3	
Java interface, java enums, java user/input, java data, java array list, java hash map, javawrapper classes, java exceptions.	**9 Hours**
Module –4	
Java file handling-java files, java create/write files, java read files, java delete files, java how to add two numbers, java keywords, java string methods, java math methods.	**9 Hours**
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) • PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules • In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule • The students will have to answer 4 full questions, selecting one full question from each module	

Text Books:
1. Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007
Reference Books:
1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, PearsonEducation,2008, ISBN:9788131720806. 2. Rajkumar Buyya,S Thamarasi selvi, xingchen chu, Object oriented Programming with java, TataMcGraw Hill education private limited. 3. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies. 4. Anita Seth and B L Juneja, JAVA One step Ahead, Oxford University Press, 2017

OBJECT ORIENTED PROGRAMMING WITH JAVA LABORATORY SEMESTER – III			
Laboratory Code	**BVOCASP203**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives:

- Gain knowledge about basic Java language syntax and semantics to write Java programs and useConcepts such as variables, conditional and iterative execution methods etc.
- Understand the fundamentals of object-oriented programming in Java, including definingclasses, objects, invoking methods etc. and exception handling mechanisms.
- Understand the principles of inheritance, packages and interfaces.

Course outcomes: The students should be able to:

CO1: Identify classes, objects, members of a class and relationships among them needed for a specificproblem

CO2: Write java application programs using oop principles and proper program structuring

CO3: Demonstrate the concepts of polymorphism and inheritance

CO4: Write java programs to implement error-handling techniques using exception handling.

CO5: Understand and apply object oriented features and java concepts

Laboratory Experiments:

1. Write a program to find factorial of list of number reading input as command line argument.
2. Write a program to display all prime numbers between two limits.
3. Write a program to sort list of elements in ascending and descending order and show the exception handling.

4. Write a program to implement Rhombus pattern reading the limit form user.
5. Write a program to implement all string operations
6. Write a program to find area of geometrical figures using method.
7. Write a program to implement constructor overloading by passing different number of Parameter of different types.
8. Write a program to create student report using applet, read the input using text boxes and display the o/p using buttons.
9. Write a program to calculate bonus for different departments using method overriding.
10. Write a program to implement thread priorities.
11. Write a program to implement thread, applets and graphics by implementing animation of ball moving.
12. Write a program to implement mouse events.
13. Write a program to implement keyboard events.
Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking four experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

PYTHON FOR DATA SCIENCE LABORATORYSEMESTER – III			
Laboratory Code	**BVOCASP204**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Explore Python language fundamentals, including basic syntax, variables, and types
- Create and manipulate regular Python lists
- Use functions and import packages
- Build Numpy arrays, and perform interesting calculations
- Create and customize plots on real data
- Supercharge your scripts with control flow, and get to know the Pandas Data Frame.

Course outcomes: The students should be able to:

CO1: Advanced concepts of python like writing python scripts.

CO2: Sequence and file operations in python.

CO3: Use libraries like pandas, NumPy, matplotlib, scikit, and master the concepts like python machine learning, scripts, and sequence

Laboratory Experiments:

1. Write a program that has a class student that stores roll number, name and marks (in three

   subjects) of the students. Display the information (roll number, name, and total marks)

storedabout the student.
2. Write a program that has classes such as Student, Course, and Department. Enroll a student in a course of a particular department
3. Write program to do the followings: a) Write a program to read two integers a and b. Compute a/b and print, when b is not zero. Raisean exception when b is equal to zero. b) Write a program that prompts the user to enter a number. If the number is Positive or Zero printit. Otherwise, raise an exception.
4. Write a NumPy program to compute sum of all elements, sum of each column and sum of each row of a given array.
5. Write a program to preprocess the data using Numpy and sklearn.preprocessing packages.
6. Write program to do the followings: a) create data frame df consisting 10 rows and 4 columns of randomly generated numbers between1 to 100 b) Create a new column such that, each row contains the row number of nearest row-record byEuclidean distance.
7. Use Automobile dataset, write program to answer the following questions: a) Clean the data and update the CSV file. b) Find the most expensive car. c) Find each company's highest price car.
8. Use Companies sales dataset, write program to answer the following questions: a) Read Total profit of all months and show it using a line plot.

b) Read all product sales data and show it using a multiline plot.

c) Calculate total sale data for last year for each product and show it using a Pie chart

9. Use SAHeart dataset, write program to answer the below questions:

a) Draw a Bar plot to show the number of person having CHD or not in comparison to they havingfamily history of the disease or not.

b) Find out the number of CHD cases in different age categories. Do a Bar Plot and sort them in theorder of age groups.

10. Use Iris data set, write program to answer the following questions:

a) Find the mean, median, standard deviation of iris's sepal length (1st Column).

b) Create a normalized form of iris's sepal length whose values range exactly between 0 and 1 sothat the minimum has value 0 and maximum has value 1.

c) Find the number and position of missing values in iris's sepal length (1st column)

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- Students allowed picking three experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

Change of experiment allowed only once and marks allotted to the procedure part to made zero.

OBJECT ORIENTED PROGRAMMING LABORATORY SEMESTER – III			
Laboratory Code	**BVOCASPCT 205**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives:

- To strengthen problem solving ability by using the characteristics of an object-oriented approach.
- To design applications using object oriented features
- To handle Exceptions in programs.
- To teach the student to implement object oriented concepts.

Course outcomes: The students should be able to:

CO1: Understand the features of C++ supporting object oriented programming

CO2: Understand the relative merits of C++ as an object oriented programming language

CO3: Understand how to produce object-oriented software using C++

CO4: Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism

CO5: Understand advanced features of C++ specifically stream I/O, templates and operator overloading

1 a) Write a C++ program to find the sum of individual digits of a positive integer.

b) Write a C++ program to generate the first n terms of the sequence.

2 a) Write a C++ program to generate all the prime numbers between 1 and n, where n is a valuesupplied by the user. b) Write a C++ program to find both the largest and smallest number in a list of integers.
3 a) Write a C++ program to sort a list of numbers in ascending order. b) Write a Program to illustrate New and Delete Keywords for dynamic memory allocation.
4 a) Write a program Illustrating Class Declarations, Definition, and Accessing Class Members. b)Program to illustrate default constructor, parameterized constructor and copy constructors c) Write a Program to Implement a Class STUDENT having Following Members: 9 Member functions Member Description assign () Assign Initial Values compute () to Compute Total, Averagedisplay () to Display the Data.
5 a) Write a Program to Demonstrate the i) Operator Overloading. ii) Function Overloading. b) Write a Program to Demonstrate Friend Function and Friend Class.
6 a) Write a Program to Access Members of a STUDENT Class Using Pointer to Object Members. b) Write a Program to Generate Fibonacci Series use Constructor to Initialize the Data Members.
7 Write a C++ program to implement the matrix ADT using a class. The operations supported by thisADT are: a) Reading a matrix. b) Addition of matrices. c) Printing a matrix. d) Subtraction of matrices. e) Multiplication of matrices
8 Write C++ programs that illustrate how the following forms of inheritance are supported:

a)Single inheritance b)Multiple inheritance c)Multi level inheritance d)Hierarchical inheritance
9 a) Write a C++ program that illustrates the order of execution of constructors and destructors whennew class derived from more than one base class. b) Write a Program to Invoking Derived Class Member Through Base Class Pointer.
10 a) Write a Template Based Program to Sort the Given List of Elements. b) Write a C++ program that uses function templates to find the largest and smallest number in a list ofintegers and to sort a list of numbers in ascending order.
11 a) Write a Program Containing a Possible Exception. Use a Try Block to Throw it and a CatchBlock to Handle it Properly. b) Write a Program to Demonstrate the Catching of All Exceptions.
Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking three experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

MOBILE APPLICATION DEVELOPMENTSEMESTER – IV			
Subject Code	**BVOCASCT 251**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Learn to setup Android application development environment
- Illustrate user interfaces for interacting with apps and triggering actions
- Interpret tasks used in handling multiple activities
- Identify options to save persistent application data
- Appraise the role of security and performance in Android applications

Course outcomes: The students should be able to:

CO1: Create, test and debug android application by setting up android development environment

CO2: Implement adaptive, responsive user interfaces that work across a wide range of devices.

CO3: Infer long running tasks and background work in android applications

CO4: Demonstrate methods in storing, sharing and retrieving data in android applications

CO5: Analyze performance of android applications and understand the role of permissions and security

CO6: Describe the steps involved in publishing android application to share with the world

Module – 1	**Teaching Hours**
ANDROID OVERVIEW, What Is Android? ,Features Of Android ,Android Applications, ANDROID – ENVIRONMENT SETUP, Step 1 - Setup Java Development Kit (JDK),Step 2 - Setup Android SDK ,Step 3 - Setup Eclipse IDE, Step 4 - Setup Android Development Tools (ADT) Plugin ,Step 5 - Create Android Virtual Device, ANDROID – ARCHITECTURE ,Linux Kernel, Libraries ,Android Runtime ,Application Framework ,Applications ANDROID – APPLICATIONS COMPONENT, Activities, Services, Broadcast Receivers, Content Providers, Additional Components, ANDROID – HELLO WORLD EXAMPLE, Create Android Application, Anatomy Of Android Application ,The Main Activity File , The Manifest File, The Strings File, The R File, The Layout File, Running The Application.	**9 Hours**
Module – 2	
Android resources, android activities, android services, android broadcast receivers, android content providers, android fragments, android intents/filters. ANDRIOD-USER INTERFACE android ui layouts, ui controls, event handling, styles and themes, custom components. Android UI DESIGN, ANDROID UI PATTERNS, ANDRIOD UI testing.	**9 Hours**
Module – 3	
Android advanced concepts-android-drag and drop, notifications, based services, sending email, sending sms, phone calls, publishing ,android alert dialogues, android animations, android audio capture, android audio manager, android auto	**9 Hours**

complete, android best practices, android Bluetooth, android camera. Android Wi-Fi, android widgets, android XML parsers.	
Module –4	
Android developer tools, android emulator, android facebook integration, android google maps, android image effects, android internal storage, android login screen, android media player, android multi touch, android navigation, android PHP/MySQL, android push notification, android SDK manager, android session management.	**9 Hours**
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) • PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules • In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule The students will have to answer 4 full questions, selecting one full question from each module	
Text Books:	
1. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley IndiaPvt Ltd, 2016. ISBN-13: 978-8126565580 2. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference",Google Developer Training Team, 2017. 3. https://www.gitbook.com/book/googledeveloper-training/android-developer-fundamentals- course-concepts/details (Download pdf file from the above link)	

Reference Books:

1. Erik Hellman, “Android Programming – Pushing the Limits”, 1st Edition, Wiley India Pvt Ltd,2014.
2. Dawn Griffiths and David Griffiths, “Head First Android Development”, 1st Edition, O’ReillySPD Publishers, 2015.
3. Anubhav Pradhan, Anil V Deshpande, “ Composing Mobile Apps” using Android, Wiley 2014,ISBN: 978-81-265-4660-2

INTRODUCTION TO GO PROGRAMMINGSEMESTER – IV			
Subject Code	**BVOCASCT252**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Learn about what makes Go a great language
- Learn how to install the Go toolchain
- Learn how to setup Visual Studio Code to edit and debug Go programs
- Learn how to work with the Go Playground to test and run snippets of Go code
- Learn and understand the basic Go language syntax and features
- Learn how to use the Go tool chain commands to compile, test, and manage Go code
- And finally, you'll learn how to work with and manage Go modules for module dependencymanagement

Course outcomes: The students should be able to:

CO1: Learn go fundamentals and apply them in real world scenarios

CO2: Get to grip with advanced features like go language and concurrency

CO3: Understand and develop your knowledge of programming fundamentals

CO4: Build up a knowledge foundation for more advanced programming languages

CO5: Learn the language behind the super popular Docker technology

CO6: Everything you need to get up and go

Module – 1	**Teaching Hours**
Introduction, Types, Variables Introduction: Getting Setup: Machine Setup, Text Editors,The Terminal, Environment, Go- Your First Program, How to Read a Go Program Types: Numbers, Strings, Booleans Variables: How to Name a Variable, Scope, Constants, Defining Multiple Variables, An Example Program.	**9 Hours**
Module – 2	
Control Structures, Arrays, Slices and Maps and Functions Control Structures: The for Statement, If Statement, Switch Statement, Arrays, Slices and Maps : Arrays, Slices-append, Copy, Maps, Functions: Your Second Function, Variadic Functions, Closure, defer, panic and recover, panic and recover, pointers, The * and & Operators, new.	**9 Hours**
Module – 3	
Structs and Interfaces, Concurrency, Packages structs and Interfaces: structs , Methods,Interfaces, Concurrency: Go routines, Channels, Packages: Creating Packages, Documentation	**9 Hours**
Module –4	
Packages and the Go Tool, Testing, Reflection Packages and the Go Tool: Introduction, Import Paths, The package Declaration. Import Declaration, Blank Imports, Packages and Naming, The Go Tool, Testing: The go test Tool, Test Functions, Coverage, Benchmark Functions, Profiling, Example Functions, Reflection: Why Reflection, reflect. Type and reflect. Value, Display, a Recursive Value Print, Example: Encoding S-E xpressions, Setting Variables with reflect.	**9 Hours**

Value, Example: DecodingS-E xpressions, Accessing Struct Field Tags , Displaying the Methods of a Type, A Word of Caution	

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule

  The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Introducing Go by Caleb Doxsey
2. An Introduction to Programming in Go By Caleb Doxsey
3. The Go Programming Language By Alan A. A. Donovan, Brian W. Kernighan

Reference Books:

1. The Way To Go, a Thorough Introduction to the Go Programming Language By IVO BALBAERT
2. Go Boot camp: Everything you need to know to get started with Go By Matt Aimonetti

References: Online Resources

1. https://golang.org/doc/code.html
2. https://gobyexample.com/
3. https://www.tutorialspoint.com/go/
4. https://www.toptal.com/go/go-programming-a-step-by-step-introductory-tutorial
5. https://learning.oreilly.com/videos/ultimate-go-programming/9780134757476

6. https://www.golangprograms.com/go-language.html
7. https://cloudacademy.com/course/introduction-go-programming-language/introduction/#:~:text=Learning%20Objectives&text=Learn%20how%20to%20setup %20Visual,test%2C%20and%20manage%20Go%20code

MOBILE APPLICATION DEVELOPMENT LABORATORY SEMESTER – IV			
Laboratory Code	**BVOCASP253**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Describe those aspects of mobile programming that make it unique from programming forother platforms,
- Critique mobile applications on their design pros and cons,
- Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces,
- Program mobile applications for the Android operating system that use basic and advancedphone features.
- Deploy applications to the Android marketplace for distribution.

Course outcomes: The students should be able to

CO1: Apply essential android programming concepts.

CO2: Develop various android applications related to layouts & rich uses interactive interfacesCO3: Gain knowledge concerning mobile operating systems and their architecture

CO4: Recognize and setup a mobile device and application runtime environment

CO5: Be able to setup programming tools for a mobile application developer (for selected

modern mobileplatforms)

CO6: Understands the need for continuous improvement of his/her skills due to the rapidly changing environment of mobile devices

Laboratory Experiments:

Part-A

1. Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax

   and the website address is to be displayed. Insert a horizontal line between the job title and the phone number.

2. Develop an Android application using controls like Button, TextView, and EditText for designing a calculator having basic functionality like Addition, Subtraction, Multiplication, andDivision.

3. Create a SIGN Up activity with Username and Password. Validation of password should happenbased on the following rules:
   - Password should contain uppercase and lowercase letters.
   - Password should contain letters and numbers.
   - Password should contain special characters.
   - Minimum length of the password (the default value is 8).

   On successful SIGN UP proceed to the next Login activity. Here the user should SIGN IN using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and after that display a toast message saying “Failed

Login Attempts" and disable the SIGN IN button. Use Bundle to transfer information from one activity to another.
4. Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.
5. Write a program to create an activity with two buttons START and STOP. On pressing of the START button, the activity must start the counter by displaying the numbers from one and the counter must keep on counting until the STOP button is pressed. Display the counter value in a Text View control.
6. Develop a simple application with one Edit Text so that the user can write some text in it. Create a button called "Convert Text to Speech" that converts the user input text into voice.
7. Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALLbutton, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.
Part-B
8. Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name.
9. Develop a content provider application with an activity called "Meeting Schedule" which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called "Meeting Info" having Date Picker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying "No Meeting on this Date".

10. Create an application to receive an incoming SMS, which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Useappropriate emulator control to send the SMS message to your application.
11. Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in MkSDcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying “First Create a File”.
12. Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move theaudio forward or backward as required.
13. Develop an application that makes use of the clipboard framework for copying and pasting ofthe text. The activity consists of two Edit Text controls and two Buttons to trigger the copy and paste functionality.
14. Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI isE = P * (r(1+r)n)/((1+r)n -1) where E = The EMI payable on the car loan

amountP = The Car loan Principal

Amount

r = The interest rate value computed on a monthly

basisn = The loan tenure in the form of months

The down payment amount has to be deducted from the principal amount paid towards buyingthe Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the Principal Amount, Down Payment, InterestRate, Loan Term (in months) and a button named as "Calculate Monthly EMI". On click of thisbutton, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan

Term and Interest Rate values.

- **Online Resource:**
- https://developer.android.com/guide
- https://www.tutorialspoint.com/android/
- https://www.udacity.com/course/android-development-for-beginners--ud837

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- Students allowed picking four experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

Change of experiment allowed only once and marks allotted to the procedure part to made zero.

INTRODUCTION TO GO PROGRAMMING LABORATORY SEMESTER – IV			
Laboratory Code	**BVOCASP254**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Master the fundamental syntax of Go, including variable declarations, data types controlstructures, and functions
- Test, build, and deploy command-line Go applications
- Benchmark your Go applications
- Master use of the Go documentation
- Manage state
- Build web services
- Write concurrent code
- Produce code more efficiently with templating
- Package and deploy complete Go applications

Course outcomes: The students should be able to

CO1: Understand the fundamentals of go programming language.

CO2: Make your own stand-alone command-line apps or scripts network and web servers.

CO3: Boost your hire ability through innovative and independent learning.

CO4: Understand and develop your knowledge of programming fundamentals CO5: Build up a knowledge foundation for more advanced programming languages
Laboratory Experiments:
1. Go Program to calculate whether a number is Even or Odd
2. Go Program to display of standard arithmetic operators with two integer values
3. Go Program to find the LCM and GCD of given two numbers
4. Go Program to find the index of first occurrence of a substring
5. Go Program to find the first and last element of slice in golang
6. Go Program to find the total numbers of characters in a string
7. Go Program to print full pyramid using STAR
8. Go Program for implementation of Binary search
9. Go Program for implementation of Linear search
10. Go Program to generate multiplication table
11. Go Program to add two matrix using multi-dimensional arrays
12. Go Program to calculate area of rectangle and square
13. Go Program to check whether a number is palindrome or not
14. Go Program to implementation of Tower of Hanoi algorithm
15. Go Program to print the ASCII code for each letter in the Alphabet
16. Go Program to read the file line by line to string
17. Go Program to take user input and addition of two strings
18. Go Program to get current date and time in various format

19. Go Program to array reverse sort functions for integer and strings
20. Go Program to replace substrings in a string
Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking three experiments from the lot.
• Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

DATA STRUCTURES LABORATORY SEMESTER – IV			
Laboratory Code	**BVOCASP255**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This laboratory course enable students to get practical experience in design,develop, implement, analyze and evaluation/testing of

- Explain fundamentals of data structures and their applications essential forprogramming/problem solving.
- Illustrate linear representation of data structures: Stack, Queues, Lists, Trees and Graphs.
- Demonstrate sorting and searching algorithms.
- Find suitable data structure during application development/Problem Solving.
- Asymptotic performance of algorithms.
- Linear data structures and their applications such as stacks, queues and lists
- Non-Linear data structures and their applications such as trees and graphs
- Sorting and searching algorithms

Course outcomes: The students should be able to:

CO1: Use different types of data structures, operations and algorithms

CO2: Apply searching and sorting operations on files

CO3: Use stack, Queue, Lists, Trees and Graphs in problem solving

CO4: Implement all data structures in a high-level language for problem solving.

CO5: Analyze and Compare various linear and non-linear data structures CO6: Code, debug and demonstrate the working nature of different types of data structures and theirapplications CO7: Implement, analyze and evaluate the searching and sorting algorithmsCO8: Choose the appropriate data structure for solving real world problems
Part-A
Introduction: Data Structures, Classifications (Primitive & Non Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, Dynamically allocated arrays. Array Operations: Traversing, inserting, deleting, searching, and sorting. Multidimensional Arrays, Polynomials and Sparse Matrices. Strings: Basic Terminology, Storing, Operations and Pattern Matching algorithms. Programming Examples.
Stacks: Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. Recursion - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Ackerman's function. Queues: Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Dequeues, Priority Queues, A Mazing Problem. Multiple Stacks and Queues. Programming Examples.
Linked Lists: Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists –

Polynomials, Sparse matrix representation. Programming Examples
Trees: Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression, Programming Examples
Part-B
1. Design, Develop and Implement a menu driven Program in C for the following array operations. a) Creating an array of N Integer Elements b) Display of array Elements with Suitable Headings c) Inserting an Element (ELEM) at a given valid Position (POS) d) Deleting an Element at a given valid Position (POS) e) Exit. Support the program with functions for each of the above operations.
2. Design, develop and Implement a Program in C for the following operations on Strings. a) Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) b) Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist inSTR Support the program with functions for each of the above operations. Don't use Built-in functions.
3. Design, Develop and Implement a menu driven Program in C for the following operations onSTACK of Integers (Array Implementation of Stack with maximum size MAX)

a) Push an Element on to Stack
b) Pop an Element from Stack
c) Demonstrate how Stack can be used to check Palindrome
d) Demonstrate Overflow and Underflow situations on Stack
e) Display the status of Stack
f) Exit

Support the program with appropriate functions for each of the above operations

4. Design, develop and Implement a Program in C for converting an Infix Expression to Postfix

   Expression. Program should support for both parenthesized and free parenthesized expressionswith the operators: +, -, *, /, % (Remainder), ^ (Power) and alphanumeric operands.

5. Design, Develop and Implement a Program in C for the following Stack Applications
   a) Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^
   b) Solving Tower of Hanoi problem with n disks

6. Design, Develop and Implement a menu driven Program in C for the following operations onCircular QUEUE of Characters (Array Implementation of Queue with maximum size MAX)
   a) Insert an Element on to Circular QUEUE
   b) Delete an Element from Circular QUEUE
   c) Demonstrate Overflow and Underflow situations on Circular QUEUE
   d) Display the status of Circular QUEUE
   e) Exit

Support the program with appropriate functions for each of the above operations

7. Design, Develop and Implement a menu driven Program in C for the following operations onSingly Linked List (SLL) of Student Data with the fields: USN, Name, Programmer, Sem, Phone
    a) Create a SLL of N Students Data by using front insertion.
    b) Display the status of SLL and count the number of nodes in it
    c) Perform Insertion / Deletion at End of SLL
    d) Perform Insertion / Deletion at Front of SLL(Demonstration of stack)
    e) Exit

8. Design, Develop and Implement a menu driven Program in C for the following operations onDoubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation,

    Sal, PhNo
    a) Create a DLL of N Employees Data by using end insertion.
    b) Display the status of DLL and count the number of nodes in it
    c) Perform Insertion and Deletion at End of DLL
    d) Perform Insertion and Deletion at Front of DLL
    e) Demonstrate how this DLL can be used as Double Ended Queue.
    f) Exit

9. Design, Develop and Implement a Program in C for the following operations on Singly CircularLinked List (SCLL) with header nodes
    a) Represent and Evaluate a Polynomial P(x,y,z) = 6x2y2z-4yz5+3x3yz+2xy5z-2xyz3
    b) Find the sum of two polynomials POLY1(x,y,z) and POLY2(x,y,z) and store the resultin POLYSUM(x,y,z)

Support the program with appropriate functions for each of the above operations

10. Design, Develop and Implement a menu driven Program in C for the following operations onBinary Search Tree (BST) of Integers.

    a) Create a BST of N Integers: 6, 9, 5, 2, 8, 15, 24, 14, 7, 8, 5, 2
    b) Traverse the BST in Inorder, Preorder and Post Order
    c) Search the BST for a given element (KEY) and report the appropriate message
    d) Exit

11. Design, Develop and Implement a Program in C for the following operations on Graph (G) ofCities

    a) Create a Graph of N cities using Adjacency Matrix.
    b) Print all the nodes reachable from a given starting node in a digraph using DFS/BFSmethod

12. Given a File of N employee records with a set K of Keys (4-digit) which uniquely determine the records in file F. Assume that file F is maintained in memory by a Hash Table (HT) of m memory locations with L as the set of memory addresses (2-digit) of locations in HT. Let the keys in K and addresses in L are Integers. Design and develop a Program in C that uses Hash function H: K →L as H(K)=K mod m (remainder method), and implement hashing technique to map a given

    key K to the address space L. Resolve the collision (if any) using linear probing

Text Books

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, UniversitiesPress, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014

Reference Books

1. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed,

CengageLearning, 2014.
2. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012.
3. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures withApplications, 2nd Ed, McGraw Hill, 2013
4. A M Tenenbaum, Data Structures using C, PHI, 1989
5. Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.

Conduction of Practical Examination:

- All laboratory experiments are to be included for practical examination.
- Students allowed picking three experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

Change of experiment allowed only once and marks allotted to the procedure part to made zero.

MOBILE APPLICATION DEVELOPMENT LABORATORY

SEMESTER –IV

Laboratory Code	BVOCASP255	**IA Marks**	30
Number of Lecture Hours/ Week	6	**Exam Marks**	120
Total number of lecture hours	72	**Exam Hours**	3

Credits - 06

Course objectives: This course will enable students to

- Describe those aspects of mobile programming that make it unique from programming for other platforms,
- Critique mobile applications on their design pros and cons,
- Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces,
- Program mobile applications for the Android operating system that use basic and advanced phone features.
- Deploy applications to the Android marketplace for distribution.

PART - A

ANDROID OVERVIEW, What Is Android?, Features Of Android ,Android Applications, ANDROID – ENVIRONMENT SETUP, Step 1 - Setup Java Development Kit (JDK), Step 2 - Setup Android SDK, Step 3 - Setup Eclipse IDE, Step 4 - Setup Android Development Tools (ADT) Plugin, Step 5 - Create Android Virtual Device, ANDROID – ARCHITECTURE, Linux Kernel, Libraries, Android Runtime, Application Framework , Applications ANDROID – APPLICATIONS COMPONENT, Activities, Services, Broadcast Receivers, Content Providers, Additional Components, ANDROID – HELLO WORLD EXAMPLE, Create

Android Application, Anatomy of Android Application, The Main Activity File, The Manifest File, The Strings File, The R File, The Layout File, Running the Application.
Android resources, android activities, android services, android broadcast receivers, android content providers, android fragments, android intents/filters. ANDRIOD-USER INTERFACE android ui layouts, ui controls, event handling, styles and themes, custom components. Android UI DESIGN, ANDROID UI PATTERNS, ANDRIOD UI testing.
Android advanced concepts-android-drag and drop, notifications, based services, sending email, sending sms, phone calls, publishing, android alert dialogues, android animations, android audio capture, android audio manager, android auto complete, android best practices, android Bluetooth, android camera. Android Wi-Fi, android widgets, android XML parsers.
Android developer tools, android emulator, android facebook integration, android google maps, android image effects, android internal storage, android login screen, android media player, android multi touch, android navigation, android PHP/MySQL, android push notification, android SDK manager, android session management.
PART – B
1. Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number.
2. Develop an Android application using controls like Button, TextView, and EditText for designing a calculator having basic functionality like Addition, Subtraction, Multiplication, and Division.

3. Create a SIGN-Up activity with Username and Password. Validation of password should happen based on the following rules: • Password should contain uppercase and lowercase letters. • Password should contain letters and numbers. • Password should contain special characters. • Minimum length of the password (the default value is 8). On successful SIGN UP proceed to the next Login activity. Here the user should SIGN IN using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and after that display a toast message saying “Failed Login Attempts” and disable the SIGN IN button. Use Bundle to transfer information from one activity to another.
4. Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.
5. Write a program to create an activity with two buttons START and STOP. On pressing of the START button, the activity must start the counter by displaying the numbers from one and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextView control
6. Develop a simple application with one EditText so that the user can write some text in it. Create a button called “Convert Text to Speech” that converts the user input text into voice.
7. Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.

8. Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name.
9. Develop a content provider application with an activity called “Meeting Schedule” which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called “Meeting Info” having Date Picker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying “No Meeting on this Date”.
10. Create an application to receive an incoming SMS, which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application.
11. Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in MkSDcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying “First Create a File”.
12. Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move the audio forward or backward as required.
13. Develop an application that makes use of the clipboard framework for copying and pasting of the text.

The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality.
14.CreateanAIDLservicethat calculates Car Loan EMI. The formula to calculate EMI is E= P * (r(1+r)n)/((1+r)n -1) where E = The EMI payable on the car loan amount P = The Car loan Principal Amount r = The interest rate value computed on a monthly basis n = The loan tenure in the form of months
The down payment amount has to be deducted from the principal amount paid towards buying the Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the Principal Amount, Down Payment, Interest Rate, Loan Term (in months) and a button named as “Calculate Monthly EMI”. On click of this button, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan Term and Interest Rate values.
Course outcomes: The students should be able to
CO1: Apply essential android programming concepts. CO2: Develop various android applications related to layouts & rich uses interactive interfaces CO3: Gain knowledge concerning mobile operating systems and their architecture CO4: Recognize and setup a mobile device and application runtime environment CO5: Be able to setup programming tools for a mobile application developer (for selected modern mobile platforms) CO6: Understands the need for continuous improvement of his/her skills due to the rapidly changing environment of mobile devices
Text Books

1. J F DiMarzio, “Beginning Android Programming with Android Studio”, 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580

2. 2. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference”, Google Developer Training Team, 2017.

3. https://www.gitbook.com/book/googledeveloper-training/android-developer-fundamentals course- concepts/details (Download pdf file from the above link)

Reference Books

1. Erik Hellman, “Android Programming – Pushing the Limits”, 1st Edition, Wiley India Pvt Ltd, 2014.

2. Dawn Griffiths and David Griffiths, “Head First Android Development”, 1st Edition, O’Reilly SPD Publishers, 2015.

3. Anubhav Pradhan, Anil V Deshpande, “Composing Mobile Apps” using Android, Wiley 2014, ISBN: 978-81-265-4660-2

Online Resource

- https://developer.android.com/guide
- https://www.tutorialspoint.com/android/
- https://www.udacity.com/course/android-development-for-beginners--ud837

Conduction of Practical Exam

1. All laboratory experiments from part B are to be included for practical examination.
2. Mini project valuated for 40 Marks and 10 Marks for record.
3. Report should be prepared in a standard format prescribed for project work.
4. Students are allowed to pick two experiment from part-B
5. Strictly follow the instructions as printed on the cover page of answer script.

Change of experiment allowed only once and marks allotted to the procedure part to made zero.

ASP .NET CORE PROGRAMMINGSEMESTER – V			
Subject Code	**BVOCASCT301**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Understand the goals and benefits of the .NET Core platform
- Learn how to build a compelling and maintainable HTML user interface using the Razor viewengine and client-side JavaScript
- Client side programming: HTTP, CGI, Cookies, JavaScript, HTML, XML.
- ASP.NET Web services and web service security.
- Introduction to the .NET framework.
- .NET Interoperation services.

Course outcomes: The students should be able to:

CO1: Learn about ms.net framework developed by Microsoft.

CO2: Be able to using xml in c#.net specifically ado.net and sql server

CO3: Be able to understand use of c# basics, objects and types, inheritanceCO4: Develop, implement and creating applications with c#.

CO5: Develop, implement, and demonstrate component services, threading, remoting, windowsservices, web

CO6: Understand and be able to explain security in the .net framework and deployment in the net.

Module – 1	**Teaching Hours**
Introduction to C# Programming: What is C#?,Does C# Replace Java?, Simple Program:Printing a Line of Text, Memory Concepts, Variables and Data types, Initialization of Variables, Variable Scope, Constants, Value Types and Reference Types, CTS Types,Operators And Statements, Arrays and Strings.	**9 Hours**
Module – 2	
Object Oriented Programming: Objects and Classes, Methods and Properties, Constructors and Destructors. **Inheritance:** Types of Inheritance, Implementation versus Interface Inheritance, MultipleInheritances, Structs and Classes, **Polymorphism:** Abstract Classes, Implementing Polymorphism by Method Overloading, Implementing Polymorphism by Method Overriding, Interfaces and Structures	**9 Hours**
Module – 3	
Interfaces: Defining and Implementing Interfaces, Derived Interfaces, Accessing Interfaces, Overriding Interfaces, **Structures:** Defining Structs, Creating Structs, Creating Enum, Exception Handling, Exception Classes, Standard Exceptions, User Defined Exceptions, Delegates	**9 Hours**
Module –4	

Introduction to ASP.NET Core: Introduction: What is ASP.NET Core?,ASP.NET Core Features, Advantages of ASP.NET Core, MVC Pattern, Understanding ASP.NET Core MVC,ASP.NET Core vs. ASP.NET MVC vs. ASP.NET Web Forms **ASP.NET Core First Application:** ASP.NET Core Environment Setup, ASP .NET Core First Application, Project Layout, Understanding Life Cycle of ASP.Net Core Request	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule

  The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Pro C# with .NET 3.0 – Andrew Troelsen, Special Edition, Dreamtech Press, India, 2007.
2. Programming in C# –E.Balagurusamy, 5th Reprint, TataMcGraw Hill, 2004. (ForProgramming Examples)
3. .Net Core in Action By Dustin Metzgar

Reference Books:

1. Inside C# – Tom Archer, WP Publishers, 2001. 2. C#:
2. The Complete Reference – Herbert Schildt, Tata McGrawHill, 2004.
3. Programming in C# A Primer third Edition- E.Balagurusamy

4. Beginning ASP.NET 2.0 in C# 2005: From Novice to Professional-by Matthew MacDonald

References: Online Resources

YOUTUBE LINK

ASP.NET tutorial for beginners

https://www.youtube.com/watch?v=3AYoipyqOkQ&list=PL6n9fhu94yhXQS_p1i-HLIftB9Y7Vnxlo **C# tutorial for beginners**

https://www.youtube.com/watch?v=SXmVym6L8dw&list=PLAC325451207E3105

https://dotnet.microsoft.com/learn/videos

Introduction to C#

https://www.youtube.com/watch?v=SXmVym6L8dw&list=PLAC325451207E3105 **Microsoft: C# fundamentals for absolute beginners**

https://www.youtube.com/watch?v=MqGM70ljpq8&list=PLyJiOytEPs4eQUuzs3PhM_7yU63jdibtf

WEBLINKS

https://dotnet.microsoft.com/learn/aspnet/what-is-aspnet-core

https://docs.microsoft.com/en-us/dotnet/core/tutorials/

https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-3.1 https://docs.microsoft.com/en-us/learn/

https://goalkicker.com/DotNETFrameworkBook/

https://www.tutorialsteacher.com/core/aspnet-core-introduction

https://www.c-sharpcorner.com/article/introduction-of-asp-net-core/

DATA STRUCTURES SEMESTER – IV			
Subject Code	**BVOCASP251**	**IA Marks**	**20**
Number of Lecture Hours/Week	**3**	**Exam Marks**	**80**
Total Number of Lecture Hours	**36**	**Exam Hours**	**3**
Credits : 03			

Course objectives: This course enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of

1. To provide the knowledge of basic data structures and their implementations.
2. To understand importance of data structures in context of writing efficient programs.
3. To develop skills to apply appropriate data structures in problem solving.

Module 1	**Teaching Hours**
Introduction: Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, Dynamically allocated arrays. Array Operations: Traversing, inserting, deleting, searching, and sorting. Multidimensional Arrays, Polynomials and Sparse Matrices. Strings: Basic Terminology, Storing, Operations and Pattern Matching algorithms. Programming Example	**9 hrs**
Module 2	
Stacks: Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. Recursion - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, Ackerman's function. Queues: Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Dequeues, Priority Queues, A Mazing Problem. Multiple Stacks and Queues. Programming Examples.	**9 hrs**

Module 3	
Linked Lists: Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Sparse matrix representation. Programming Examples.	**9 hrs**
Module 4	
Trees: Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Application of Trees-Evaluation of Expression, Programming Examples.	**9 hrs**

Course outcomes: The students should be able to:

Upon Completing the Course, Students will able to:

1. Learn the basic types for data structure, implementation and application.
2. Know the strength and weakness of different data structures.
3. Use the appropriate data structure in context of solution of given problem.
4. Develop programming skills which require solving given problem.

Question paper pattern

1. The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
2. PART-A will have 12 questions covering all four modules the students must answer 10 questions
3. PART-B will have total eight questions covering all four modules
4. In Part-B there will be 2 full questions (with a maximum of three sub questions) from each module the students will have to answer 4 full questions, selecting one full question from each module

Text Books

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014

3. Systamatic Approach to Data Structures using C, A.M Padma Reddy.
Reference Books
1. Gilberg & Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning, 2014 **2.** Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012. **3.** Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013 **4.** A M Tenenbaum, Data Structures using C, PHI, 1989 **5.** Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.

DATA STRUCTURES LABORATORY			
SEMESTER – IV			
Subject Code	**BVOCASP253**	**IA Marks**	**30**
Number of Lecture Hours/Week	**6**	**Exam Marks**	**120**
Total Number of Lecture Hours	**72**	**Exam Hours**	**3**
Credits - 06			

Course objectives: This course enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of

- Explain fundamentals of data structures and their applications essential for programming/problem solving.
- Illustrate linear representation of data structures: Stack, Queues, Lists, Trees and Graphs.
- Demonstrate sorting and searching algorithms.
- Find suitable data structure during application development/Problem Solving.
- Asymptotic performance of algorithms.
- Linear data structures and their applications such as stacks, queues and lists Non-Linear data structures and their applications such as trees and graphs
- Sorting and searching algorithms

1. Design, Develop and Implement a menu driven Program in C for the following array operations.

a) Creating an array of N Integer Elements

b) Display of array Elements with Suitable Headings

c) Inserting an Element (ELEM) at a given valid Position (POS)

d) Deleting an Element at a given valid Position (POS)

e) Exit.

Support the program with functions for each of the above operations.

2. Design, develop and Implement a Program in C for the following operations on Strings. a) Read a main String (STR), a Pattern String (PAT) and a Replace String (REP) b) Perform Pattern Matching Operation: Find and Replace all occurrences of PAT in STR with REP if PAT exists in STR. Report suitable messages in case PAT does not exist in STR Support the program with functions for each of the above operations. Don't use Built-in functions
3. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX) a) Push an Element on to Stack b) Pop an Element from Stack c) Demonstrate how Stack can be used to check Palindrome d) Demonstrate Overflow and Underflow situations on Stack e) Display the status of Stack f) Exit Support the program with appropriate functions for each of the above operations.
4. Design, develop and Implement a Program in C for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, % (Remainder), ^ (Power) and alphanumeric operands.
5. Design, Develop and Implement a Program in C for the following Stack Application to Evaluate the Suffix expression with single digit operands and operators: +, -, *, /, %,
6. Design, Develop and Implement a Program in C for the following Stack Application to Solve the problem of Tower of Hanoi with n disks.
7. Design, Develop and Implement a menu driven Program in C for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX) a) Insert an Element on to Circular QUEUE b) Delete an Element from Circular QUEUE c) Demonstrate Overflow and Underflow situations on Circular QUEUE d) Display the status of Circular QUEUE e) Exit Support the program with appropriate functions for each of the above operations.

8. Design, Develop and Implement a menu driven Program in C for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Programmer, Sem, Phone a) Create a SLL of N Students Data by using front insertion. b) Display the status of SLL and count the number of nodes in it c) Perform Insertion / Deletion at End of SLL d) Perform Insertion / Deletion at Front of SLL (Demonstration of stack) e) Exit
9. Design, Develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo a) Create a DLL of N Employees Data by using end insertion. b) Display the status of DLL and count the number of nodes in it c) Perform Insertion and Deletion at rear of DLL d) Perform Insertion and Deletion at Front of DLL e) Demonstrate how this DLL can be used as Double Ended Queue. f) Exit

MACHINE LEARNING WITH PYTHONSEMESTER – V			
Subject Code	**BVOCASCT302**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Conceptualization and summarization of big data and machine learning,
- Trivial data versus big data,
- Big data computing technologies,
- Machine learning techniques, and scaling up machine learning approaches.

Course outcomes: The students should be able to:

CO1: Show an ability to identify the characteristics of datasets and compare the trivial data and big datafor various applications.

CO2: Exhibit an Ability to select and implement machine learning techniques and computing environment that are suitable for the applications under consideration.

CO3: Demonstrate an ability to solve problems associated with batch learning and online learning, and the big data characteristics such as high dimensionality, dynamically growing data and in particular scalability issues.

CO4: Understand and apply scaling up machine learning techniques and associated computing techniques and technologies.

CO5: Recognize and implement various ways of selecting suitable model parameters for different machine learning techniques.

CO6: Be able to integrate machine learning libraries and mathematical and statistical tools with

modern technologies like Hadoop and map reduce	
Module – 1	**Teaching Hours**
Introduction: Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning. Concept Learning: Concept learning task, Concept learning as search, Find-S algorithm, Version space, Candidate Elimination algorithm, Inductive Bias.	**9 Hours**
Module – 2	
Linear Regression: Simple Linear Regression, steps in building a Regression model, Building Simple Linear Regression Model, Model Diagnostics, Multiple Linear Regression.	**9 Hours**
Module – 3	
Classification: Classification Overview, Binary Logistic Regression, Credit Classification,Gain Chart and Lift Chart, Classification tree. Bayesian Learning: Introduction, Bayes theorem, Bayes theorem and concept learning.	**9 Hours**
Module –4	
Advanced Machine Learning: Gradient Descent Algorithm, Scikit-Learn Library forMachine Learning. **Clustering:** Finding similarity distance, K-Means Clustering, Creating Product Segments	**9 Hours**

using Clustering, Hierarchical Clustering.	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) • PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules	
• In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule The students will have to answer 4 full questions, selecting one full question from each module	
Text Books:	
1. Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education. 2. Manaranjan Pradhan, U Dinesh Kumar, Machine Learning using Python, Wiley Publication.	
Reference Books:	
1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2ndedition, springer series in statistics. 2. Ethem Alpaydın, Introduction to machine learning, second edition, MIT press.	

UNIX AND SHELL PROGRAMMINGSEMESTER – V			
Subject Code	**BVOCASCT303**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to
To provide introduction to UNIX Operating System and its File System
To gain an understanding of important aspects related to the SHELL and the process
Demonstrate the working of basic commands of Unix environment including file processing
Demonstrate the usage of different shell commands, variable and AWK filtering to the given problem
To provide a comprehensive introduction to SHELL programming, services and utilities

Course outcomes: The students should be able to:

CO1: Describe the architecture and features of Unix operating system and distinguish it from otheroperating system

CO2: Demonstrate Unix commands for file handling and process control

CO3: Write regular expressions for pattern matching and apply them to various filters for a specific task

CO4: Analyze a given problem and apply requisite facets of shell programming in order to devise a shellscript to solve the problem.

Module – 1	**Teaching Hours**
The Unix Operating System, The UNIX architecture and Command Usage, The File System, Basic File Attributes, the vi Editor.	**9 Hours**

Module – 2	
The Shell, The Process, Customizing the environment, More file attributes, Simple filters	**9 Hours**
Module – 3	
Filters using regular expressions, Essential Shell Programming	**9 Hours**
Module –4	
awk – An Advanced Filter, perl - The Master Manipulator	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule

The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Sumitabha Das: UNIX – Concepts and Applications, 4th Edition, Tata McGraw Hill, 2006. (Chapters 1.2, 2, 4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19)

Reference Books:

1. Behrouz A. Forouzan and Richard F. Gilberg: UNIX and Shell Programming, Cengage Learning,2005.
2. M.G. Venkateshmurthy: UNIX & Shell Programming, Pearson Education, 2005.

MACHINE LEARNING WITH PYTHON LABORATORYSEMESTER – V			
Laboratory Code	**BVOCASP305**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- The objectives of the course "**Machine Learning and Data Mining**" is to introduce students tostate-of-the-art methods and modern programming tools for data analysis.

Course outcomes: The students should be able to:

CO1: Understand complexity of machine learning algorithms and their limitations;

CO2: Understand modern notions in data analysis oriented computing

CO3: Be capable of confidently applying common machine learning algorithms in practice and implementing their own;

CO4: Be capable of performing distributed computations;

CO5: Be capable of performing experiments in machine learning using real-world data

Laboratory Experiments:

1. Fashion Trends Online (FTO) is an e-commerce company that sells women apparel. It is observed that 10% of their customers return the items purchased by them for many reasons (such as size, color and material mismatch). On a specific day 20 customers purchased items from FTO. Write program to answer the following:
   A) Probability that exactly five customers will return the items.
   B) Probability that a maximum of five customers will return the items.
   C) Probability that more than five customers will return the items purchased by them.
   D) Average number of customers who are likely to return the items and the variance andthe standard deviation of the number of returns.

2. The number of calls arriving at a call center follows a passion distribution at 10 calls per hour.Write program to answer the following.
   A) Calculate the probability that the number of calls will be maximum five.
   B) Calculate the probability that the number of calls over a 3 hour period will exceed 30.

3. As per survey of pesticides among 1000 farmers in grape farming for around 10 acres of grape farmland, it was found that the grape farmers spray 38 liters of pesticides in a week on an average with the corresponding standard deviation of 5 liters. Assume that the pesticide spray per week follows a normal distribution. Write program to answer the following questions:
   A) What proportion of the farmers is spraying more than 50 liters of pesticide in a week?
   B) What proportion of the farmers is spraying less than 10 liters?
   C) What proportion of the farmers is spraying between 30 liters and 60 liters?

4. Implement and demonstrate the **FIND-S algorithm** for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a CSV file.

5. For a given set of training data examples stored in a .CSV file, implement and demonstrate the **Candidate-Elimination algorithm** to output a description of the set of all hypotheses consistentwith the training examples.
6. Design a program to implement the **Simple Linear Regression Model** for a sample training dataset stored as a CSV file.
7. Design a program to implement the **Multi Linear Regression Model** for a sample training data set stored as a CSV file.
8. Design a program to implement the **Bayesian classifier** for a sample training data set stored as a CSV file.
9. Write a program to implement the **Gradient Descent** algorithm for predicting future sales using the dataset Advertising.csv.
10. Design a program to implement the **K-Means Clustering** for a sample training data set stored as a CSV file
Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking three experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

UNIX AND SHELL PROGRAMMING LABORATORY SEMESTER – V			
Laboratory Code	**BVOCASP306**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- This course introduces basic understanding of UNIX OS, UNIX commands and File system and to familiarize students with the Linux environment. To make student learn fundamentals of shell scripting and shell programming. Emphases are on making student familiar with UNIX environment and issues related to it.

Course outcomes: The students should be able to:

CO1: Demonstrate the working of basic commands of unix environment including file processing

CO2: Apply regular expression to perform pattern matching using utilities like grep, sed and awk.

CO3: Implement unix commands/ system calls to demonstrate process management

CO4: Demonstrate the usage of different shell commands, variable and awk filtering to the given problem.

CO5: Develop shell scripts for developing the simple applications to the given problem

Laboratory Experiments:

Laboratory Experiments:

a) Explore Unix Environment.

b) Explore vi- editor with Vim tutor.

Perform the following operations using vi editor, but not limited to: 1. Insert character, delete character, and replace character. 2. Save File and continue working. 3. Save File and exit editor. 4. Quit the editor. 5. Quit without saving the file. 6. Rename a file. 7. Insert lines, delete line. 8. Setline numbers. 9. Search for a pattern. 10. Move forward and backward.
1a. Write a shell script that takes a valid directory name as an argument recursively descend all the sub-directors, find the maximum length of any file in that hierarchy and writ the maximum value tothe standard output. 1b. Write a shell script that accepts a path name and creates all the components in that path name as directories. For example, if the script is named as mpc, then the command mpc a/b/c/d should createsub-directories a, a/b, a/b/c, a/b/c/d.
2a. Write a shell script that accepts two filenames as arguments, checks if the permissions for these files are identical and if the permissions are identical, output common permissions otherwise outputeach filename followed by its permissions. 2b. Write a shell script which accepts valid log-in names as arguments and prints their correspondinghome directories, if no arguments are specified, print a suitable error message.

3a. Create a script file called file properties that reads a filename entered and outputs it properties. 3b. Write a shell script to implement terminal locking (Similar to the lock command). It should prompt for the user for a password. After accepting the password entered by the user, it must prompt again for the matching password as confirmation and if match occurs, it must lock the keyword until a matching password is entered again by the user. Note the Script must be written to disregard BREAK, control-D. No time limit need be implemented for the lock duration
4a. Write a shell script that accept one or more file names as argument and convert all of them to uppercase, provided they exists in current directory.
4b. Write a shell script that displays all the links to a file specified as the first argument to the script. The second argument, which is optional, can be used to specify in which the search is to begin. If this second argument is not present, the search is to begin in the current working directory. In either case, the starting directory as well as its subdirectories at all levels must be searched. The script neednot include error checking.
5a. Write a shell script that accepts filename as argument and display its creation time if file existand if does not send output error message. 5b. Write a shell script to display the calendar for the current month with current date replaced by *or ** depending whether the date is one digit or two digit.
6a. Write s a shell script to find a file/s that matches a pattern given as command line argument inthe home directory, display the contents of the file and copy the file into the directory ~/mydir. 6b. Write a shell script to list all the files in a directory whose filename is at least 10 characters. (Useexpr command to check the length).

7a. Write a shell script that gets executed and displays the message either “Good Morning” or “GoodAfternoon” or “Good Evening” depending upon time at which the user logs in. 7b. Write a shell script that accepts a list of filenames as its argument, count and report occurrenceof each word that is present in the first argument file on other argument files
8a. Write a shell script that determine the period for which as specified user is working on a systemand display appropriate message. 8b. Write a shell script that reports the logging on of as specified user within one minute after he/she login. The script automatically terminates if specified user does not login during specified in periodof time.
9a. Write a shell script that accepts the filename, starting and ending line number as an argument anddisplay all the lines between the given line number. 9b. Write a shell script that folds long lines into 40 columns. Thus any line that exceeds 40 charactersmust be broken after 40th, a “/” is to be appended as the indication of folding and processing is to be continued with the residue. The input is to be supplied through a text file created by the user.
10a. Write an awk script that accepts date argument in the form of dd-mm-yy and display it in the form month, day and year. The script should check the validity of the argument and in the case of error, display a suitable message. 10b. Write an awk script to delete duplicated line from a text file. The order of the original lines mustremain unchanged.

11a. Write an awk script to find out total number of books sold in each discipline as well as totalbook sold using associate array down table as given below. Electrical-34 Mechanical-67 Electrical-80 Computer Science-43Civil-98 Mechanical-65 Computer Science-64
11b. Write an awk script to compute gross salary of an employee accordingly to rule given below. If basic salary < 10000 then HRA=15% of basic & DA=45% of basic. If basic salary is >=1000 then HRA=20% of basic & DA=50% of basic.
Conduction of Practical Examination:
• All laboratory experiments are to be included for practical examination. • Students allowed picking three experiments from the lot. • Strictly follow the instructions as printed on the cover page of answer script **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

DEVOPS FOR DEVELOPERSSEMESTER – V			
Subject Code	**BVOCASCT304**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

Understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet your business requirements

Be able to deliver change requests from customers rapidly and effectively by adding new (and updating existing) features

Obtain complete knowledge of the "version control system" to effectively track changes augmented with Git and Github training

Have a detailed overview of continuous integration and container ecosystem by learning tools such as Jenkins and Docker

Successfully transition from a software engineer to a DevOps engineer

Course outcomes: The students should be able to:

CO1: Be able to continuous development, continuous testing, and configuration management, includingcontinuous integration and continuous deployment and finally continuous monitoring of the software

throughout its devops development life cycle.

Module – 1	**Teaching Hours**
Fundamentals: Beginning DevOps for Developers, Introducing DevOps, Building Blocksof DevOps	**9 Hours**

Module – 2	
Metrics and Measurement View: Quality and Testing, Introduce Shared Incentives	**9 Hours**
Module – 3	
Process View: Gain fast Feedback, Unified and Holistic Approach	**9 Hours**
Module –4	
Technical View: Automatic Releasing, Infrastructure as code, Specification by Example	**9 Hours**
Question paper pattern:	
The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) PART-A will have 12 questions covering all four modules the students must answer 10 questions PART-B will have total eight questions covering all four modules In Part-B there will be 2 full questions (with a maximum of three sub questions) from each module ıe students will have to answer 4 full questions, selecting one full question from each module	
Text Books:	
1. DevOps for Developers -Integrate Development and Operations, The Agile Way By Michael Huttermann	
Reference Books:	
1. The DevOps Handbook-How to create World-Class Agility, Reliability, & Security in Technology Organization By Gene Kim, Jez Humble, Patrick Debois, and John Willis	

DEVOPS FOR DEVELOPERS LABORATORYSEMESTER – V			
Laboratory Code	**BVOCASP307**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

- Understand the fundamentals of DevOps engineering and be fully proficient with DevOpsterminologies, concepts, benefits, and deployment options to meet your business requirements
- Be able to deliver change requests from customers rapidly and effectively by adding new (andupdating existing) features
- Obtain complete knowledge of the "version control system" to effectively track changesaugmented with Git and Github training
- Have a detailed overview of continuous integration and container ecosystem by learning toolssuch as Jenkins and Docker
- Successfully transition from a software engineer to a DevOps engineer

Course outcomes: The students should be able to:

CO1: Continuous development, continuous testing, configuration management, including continuous integration and continuous deployment and finally continuous monitoring of the software throughout itsDevOPs development life cycle.

Laboratory Experiments:

1. Introduction to DevOps environment: Why DevOps? What is DevOps? DevOps Market Trends,DevOps Engineer Skills, DevOps Delivery Pipeline, DevOps Ecosystem after understanding the

concepts each individual has to create / find out at least two Use Case why DevOps is required?
2. Version Control with Git: What is version control, What is Git, Why Git for your organization, Install Git, Common commands in Git, Working with Remote Repositories
3. Continuous Integration using Jenkins: Jenkins Management, Adding a slave node to Jenkins,Building Delivery Pipeline, Pipeline as a Code, Build the pipeline of jobs using Jenkins, Create a pipeline script to deploy an application over the tomcat server.
4. Configuration Management with Ansible: Introduction to Ansible, Ansible Installation, Configuring Ansible Roles, Write Playbooks, Executing adhoc command
5. Introduction to DevOps on Cloud: Learn about various cloud services and service providers, also get the brief idea of how to implement DevOps using AWS. Why Cloud?, Introduction to Cloud Computing, Why DevOps on Cloud?, Introduction to AWS, Various AWS services, DevOps using AWS
Conduction of Practical Examination:
• Students has to conduct a mini project on laboratory experiments considering two different testcases related to DevOps • Use open Source tools for successful completion of mini project • Refer to online resources to complete the project • All laboratory experiments should be carried out by each individual student • Strictly follow the instructions as printed on the cover page of answer script • Students has to build Two different Test Cases / Mini Project by considering the lab experimentsTest Case 1: 45 Marks , Test Case 2: 45 Marks, Viva-Voice: 10 Marks, Record: 20 Marks **Change of experiment allowed only once and marks allotted to the procedure part to made zero.**

INTERNET OF THINGSSEMESTER – VI			
Subject Code	**BVOCASCT351**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Assess the genesis and impact of IoT applications, architectures in real world.
- Illustrate diverse methods of deploying smart objects and connect them to network.
- Compare different Application protocols for IoT.
- Infer the role of Data Analytics and Security in IoT.
- Identify sensor technologies for sensing real world entities and understand the role of IoT invarious domains of Industry.

Course outcomes: The students should be able to:

CO1: Interpret the impact and challenges posed by IoT networks leading to new architectural models.

CO2: Compare and contrast the deployment of smart objects and the technologies to connect them tonetwork.

CO3: Appraise the role of IoT protocols for efficient network communication.

CO4: Elaborate the need for data analytics and security in IoT

CO5: Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in industry.

Module – 1	**Teaching Hours**

What is IoT, Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and IoT, IoT Challenges, IoT Network Architecture and Design, Drivers Behind New Network Architectures, Comparing IoT Architectures, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack? Smart Objects: The "Things" in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies.	**9 Hours**
Module – 2	
IP as the IoT Network Layer, The Business Case for IP, The need for Optimization, Optimizing IP for IoT, Profiles and Compliances, Application Protocols for IoT, TheTransport Layer, IoT Application Transport Methods.	**9 Hours**
Module – 3	
Data and Analytics for IoT, An Introduction to Data Analytics for IoT, Machine Learning, Big Data Analytics Tools and Technology, Edge Streaming Analytics, Network Analytics, Securing IoT, A Brief History of OT Security, Common Challenges in OT Security, How IT and OT Security Practices and Systems Vary, Formal Risk Analysis Structures: OCTAVE and FAIR, The Phased Application of Security in an Operational Environment	**9 Hours**
Module –4	

IoT Physical Devices and Endpoints - Arduino UNO: Introduction to Arduino, Arduino UNO, Installing the Software, Fundamentals of Arduino Programming. IoT Physical Devices and Endpoints - RaspberryPi: Introduction to RaspberryPi, About the RaspberryPi Board: Hardware Layout, Operating Systems on RaspberryPi, Configuring RaspberryPi, Programming RaspberryPi with Python, Wireless Temperature Monitoring System Using Pi, DS18B20 Temperature Sensor, Connecting Raspberry Pi via SSH, Accessing Temperature from DS18B20 sensors, Remote access to RaspberryPi, Smart and Connected Cities, An IoT Strategy for Smarter Cities, Smart City IoT Architecture, Smart City Security Architecture, Smart City Use-Case Examples.	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions
- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule

The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1st Edition, Pearson Education (Cisco Press Indian Reprint). (ISBN: 978-9386873743)
2. Srinivasa K G, "Internet of Things", CENGAGE Leaning India, 2017

Reference Books:
1. Vijay Madisetti and ArshdeepBahga, “Internet of Things (A Hands-on-Approach)”, 1st EditionVPT, 2014. (ISBN: 978-8173719547) 2. Raj Kamal, “Internet of Things: Architecture and Design Principles”, 1st Edition, McGraw HillEducation, 2017. (ISBN: 978-9352605224)
Possible list of practicals:
1. Transmit a string using UART 2. Point-to-Point communication of two Motes over the radio frequency. 3. Multi-point to single point communication of Motes over the radio frequency. LAN(Subnetting). 4. I2C protocol study 5. Reading Temperature and Relative Humidity value from the sensor

COMPUTER NETWORKSSEMESTER – VI			
Subject Code	**BVOCASCT352**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Demonstration of application layer protocols
- Discuss transport layer services and understand UDP and TCP protocols
- Explain routers, IP and Routing Algorithms in network layer
- Disseminate the Wireless and Mobile Networks covering IEEE 802.11 Standard
- Illustrate concepts of Multimedia Networking, Security and Network Management

Course outcomes: The students should be able to:

CO1: Explain principles of application layer protocols

CO2: Recognize transport layer services and infer udp and tcp protocols

CO3: Classify routers, ip and routing algorithms in network layer

CO4: Understand the wireless and mobile networks covering ieee 802.11 standard

CO5: Describe multimedia networking and network management

Module – 1	**Teaching Hours**

Application Layer: Principles of Network Applications: Network Application Architectures, Processes Communicating, Transport Services Available to Applications, Transport Services Provided by the Internet, Application-Layer Protocols. The Web and HTTP: Overview of HTTP, Non-persistent and Persistent Connections, HTTP Message Format, User-Server Interaction: Cookies, Web Caching, The Conditional GET, File Transfer: FTP Commands & Replies, Electronic Mail in the Internet: SMTP, Comparison with HTTP, Mail Message Format, Mail Access Protocols, DNS; The Internet's Directory Service: Services Provided by DNS, Overview of How DNS Works, DNS Records and Messages, Peer-to-Peer Applications: P2P File Distribution, Distributed Hash Tables.	**9 Hours**
Module – 2	
Transport Layer : Introduction and Transport-Layer Services: Relationship Between Transport and Network Layers, Overview of the Transport Layer in the Internet, Multiplexing and Demultiplexing: Connectionless Transport: UDP,UDP Segment Structure, UDP Checksum, Principles of Reliable Data Transfer: Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocols, Go-Back-N, Selective repeat, Connection-Oriented Transport TCP: The TCP Connection, TCP Segment Structure, Round-Trip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, Principles of Congestion Control: The Causes and the Costs of Congestion, Approaches to Congestion Control, Network-assisted congestion-control example, ATM ABR Congestion control, TCP Congestion Control: Fairness.	**9 Hours**

Module – 3	
The Network layer: What's Inside a Router? Input Processing, Switching, Output Processing, Where Does Queuing Occur? Routing control plane, IPv6,A Brief foray into IP Security, Routing Algorithms: The Link-State (LS) Routing Algorithm, The Distance- Vector (DV) Routing Algorithm, Hierarchical Routing, Routing in the Internet, Intra-AS Routing in the Internet: RIP, Intra-AS Routing in the Internet: OSPF, Inter/AS Routing: BGP, Broadcast Routing Algorithms and Multicast.	**9 Hours**
Module –4	
Multimedia Networking: Properties of video, properties of Audio, Types of multimedia Network Applications, Streaming stored video: UDP Streaming, HTTP Streaming, Adaptive streaming and DASH, content distribution Networks, case studies: : Netflix, You Tube and Kankan. Network Support for Multimedia: Dimensioning Best-Effort Networks, Providing Multiple Classes of Service, Diffserv, Per-Connection Quality-of Service (QoS)Guarantees: Resource Reservation and Call Admission	**9 Hours**
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) • PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules • In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule The students will have to answer 4 full questions, selecting one full question from each module	

Text Books:
1. James F Kurose and Keith W Ross, Computer Networking, A Top-Down Approach, Sixth edition, Pearson,2017 .
Reference Books:
1. Behrouz A Forouzan, Data and Communications and Networking, Fifth Edition, McGraw Hill,Indian Edition 2. Larry L Peterson and Brusce S Davie, Computer Networks, fifth edition, ELSEVIER 3. Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson 4. Mayank Dave, Computer Networks, Second edition, Cengage Learning

ETHICAL HACKING TECHNIQUESSEMESTER – VI			
Subject Code	**BVOCASCT353**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Knowledge about Ethical Hacking.
- Understand the basic concepts of Open Source Intelligence.
- Analyze the Hacking Techniques by developing simple tools

Course outcomes: The students should be able to:

CO1: Gain knowledge about ethical hacking and penetration testing.

CO2: Learn about various types of attacks, attackers and security threats and vulnerabilities present inthe computer system.

CO3: Examine how social engineering can be done by attacker to gain access of useful & sensitiveinformation about the confidential data

CO4: Learn about cryptography, and basics of web application attacks.

CO5: Gain knowledge of the tools, techniques and ethical issues likely to face the domain of ethicalhacking and ethical responsibilities.

Module – 1	**Teaching Hours**
Prerequisites: Cyber Crime Investigation and Digital Laws **Introduction to Ethical Hacking:** Requirements to get started to enter into	**9 Hours**

worlds ofHacking – Methodology of Ethical Hacking – Ethical Hacking Process.	
Module – 2	
Cyber Issues: Window Password Hacking and Cracking – Steganography - Hiding Secret Message - Anonymous Call, Message and Email Header Analysis - Access Darknet or Darkweb Using TOR : Anonymous Browsing - Access Darknet or Darkweb Using TOR : Anonymous Browsing	**9 Hours**
Module – 3	
Malware and Keylogger Analysis: Malware Analysis and Investigation – Introduction to Malware – Static Malware Analysis - Mobile Phone Hacking & Penetration Testing -Introduction of Keylogger : Art of Spying	**9 Hours**
Module –4	
Ethical Hacking Lab Setup: Cyber Security and Penetration Testing Lab - Learn Basics of Kali Linux : Hackers Operating System - Metasploit Extreme on Kali Linux : Hacking Windows 7,8,10 Like Blackhat - Bug Bounty Hunting : Web Application Penetration Testing	**9 Hours**
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks) • PART-A will have 12 questions covering all four modules the students must answer 10questions • PART-B will have total eight questions covering all four modules • In Part-B there will be 2 full questions (with a maximum of three sub questions)	

from eachmodule The students will have to answer 4 full questions, selecting one full question from each module
Text Books:
1. Charles P. Pfleeger Shari Lawrence Pfleeger Jonathan Margulies, Security in Computing, 5th Edition , Pearson Education , 2015
Reference Books:
1. Martti Lehto, Pekka Neittaanmäki, Cyber Security: Analytics, Technology and Automationedited, Springer International Publishing Switzerland, 2015 2. E Book: https://www.newhorizons.com/promotions/cybersecurity-ebooks 3. MOOC: https://www.mooc-list.com/course/penetration-testing-and-ethical-hacking-cybrary

NOSQL DATABASESEMESTER – VI			
Subject Code	**BVOCASCT354**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 02			

Course objectives: This course will enable students to

- Define, compare and use the four types of NoSQL Databases (Document-oriented, KeyValuePairs, Column-oriented and Graph).
- Demonstrate an understanding of the detailed architecture, define objects, load data, query dataand performance tune Column-oriented NoSQL databases.
- Explain the detailed architecture, define objects, load data, query data and performance tuneDocument-oriented NoSQL databases.

Course outcomes: The students should be able to:

CO1: Define, compare and use the four types of NoSQL databases (document-oriented, KeyValue pairs,column-oriented and graph).

CO2: Demonstrate an understanding of the detailed architecture, define objects, load data, query dataand performance tune column-oriented NoSQL databases.

CO3: Explain the detailed architecture, define objects, load data, query data and performance tune document-oriented NoSQL databases

Module – 1	**Teaching Hours**
Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, A (Mostly) Standard Model, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of	**9 Hours**

NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing Aggregate-Oriented Databases. More Details on Data Models; Relationships, Graph Databases, Schemaless Databases, Materialized Views, Modeling for Data Access.	
Module – 2	
Distribution Models; Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication. Consistency, Update Consistency, Read Consistency, Relaxing Consistency, The CAP Theorem, Relaxing Durability, Quorums. Version Stamps, Business and System Transactions, Version Stamps on Multiple Nodes	**9 Hours**
Module – 3	
Map-Reduce, Basic Map-Reduce, Partitioning and Combining, Composing Map-Reduce Calculations, A Two Stage Map-Reduce Example, Incremental Map-Reduce Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets	**9 Hours**
Module –4	
Introduction to MongoDB, Installing MongoDB, The Data Model, Working with Data, Advanced Queries, Database Administration, Replication, Sharding	**9 Hours**

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (60 Marks)
- PART-A will have 12 questions covering all four modules the students must answer 10questions

- PART-B will have total eight questions covering all four modules
- In Part-B there will be 2 full questions (with a maximum of three sub questions) from eachmodule

The students will have to answer 4 full questions, selecting one full question from each module

Text Books:

1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of PolyglotPersistence, Pearson Addision Wesley, 2012
2. The Definitive Guide to MongoDB, By Davud Hows, Peter Membrey, Eelco Plugge, TimHawkins, Third Edition.

Reference Books:

1. Dan Sullivan, "NoSQL for Mere Mortals", 1st Edition, Pearson Education India, 2015. (ISBN13: 978-9332557338)
2. Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A guide for Managers and the Rest of us", 1st Edition, Manning Publication/Dreamtech Press, 2013. (ISBN-13: 978- 9351192022)
3. Kristina Chodorow, "Mongodb: The Definitive Guide- Powerful and Scalable Data Storage",2nd Edition, O'Reilly Publications, 2013. (ISBN-13: 978-9351102694)
4. Practical MongoDB, By Shakuntala Gupta Edward, Navin Sabharwal.(ISBN-13 (pbk): 978-1-4842-0648-5)
5. Professional NoSQL, By Shashank Tiwari.(ISBN: 978-0-470-94224-6)

BASIC MATHEMATICS

UNIT – I

Algebra: Logarithm – Introduction, Definition, Laws of operations, Change of base, Permutations and combinations - Definitions, Related problems.

Analytical geometry – Introduction, Distance between two points, Section formula, External division, coordinates of centroid, Area of a triangle,Straight line- Slope of a straight line, Different forms of equations of the straight line, Related problems.

UNIT – II

Trigonometry: Introduction, Measurement of angles, Trigonometric functions, Relation between trigonometric functions, Signs of trigonometric functions, Trigonometric functions of standard angles. **Calculus**: Limit of functions, Definition, Formulas, Related problems. **Differentiation**: Definition of derivative of a function of one variable, Formula of standard functions, Algebra of derivative of functions, related problems**. Integration**: Definition, Formula of integrations and related problems, Indefinite integrals, Definite integrals and problems.

UNIT – III

Logical statements and truth table: Introduction, Definition, truth tables, negation, Compounding, Negation of compound statement, Tautologies and Fallacies, Prepositions, Algebra of Prepositions, Conditional statements, Biconditional statements.

Set theory: Sets, Union, Intersection, The Power set, Venn diagrams, Cartesian products. **Relations**: Relations, Properties of binary relations in set, Equivalence relations. **Functions**: Definition and Introduction, Composition of functions, Inverse functions.

UNIT – IV

Matrix Algebra: Introduction, definition, Types of matrices, Scalar multiplication of matrices, Equality of matrices, Matrix operations, Addition and subtraction, Multiplication, Transpose of a

matrix, Determinants of a square matrix, Determinant of order two, Determinant of order three, Minors of a matrix, Co-factors of a matrix, Adjoint of a square matrix, Inverse of a matrix, Rank of matrix.

Reference Books:

1. D.C. Senchethi and V.K. Kapoor, Business Mathematics, 11th edition, Sulthan Chand and Sons.
2. Prof T. Prakash Prabhu, A classic Text book of Mathematics for I PUC, 3rd edition, SDM Excellent publication, Ujire.
3. Prof T. Prakash Prabhu, A classic Text book of Mathematics for II PUC, 1st edition, SDM Excellent publication, Ujire.
4. Padmalochan Hazarika, A Text book of Business Mathematics, 2nd Edition, S. Chand Publishing.

PROJECT WORKSEMESTER – VI			
Subject Code	**BVOCASCT355**	**IA Marks**	90
Number of Lecture Hours/Week		**Exam Marks**	360
Total Number of Lecture Hours		**Exam Hours**	03
CREDITS – 09			

Course Outcomes:

CO1: Have the ability to obtain and use the mathematical, scientific and engineering-based knowledgetowards an in-depth technical efficacy in the field of software development.

CO2: Have the ability to identify, conclude and resolve software development related issues.

CO3: Able to design a system, component or process to fulfill the needs in the actual constraints likesurroundings, community, and ethic and cyber security.

CO4: Understand and be determined towards professional responsibility and ethics.

CO5: Have the ability to design and conduct experiments, as well as analyze and translate data.

CO6: Have the ability to use the method, skills and modern software development equipment in softwareengineering practices.

CO7 Have the ability to function effectively as individuals and group members, along with the abilityto lead and manage.

CO8: Have the ability to identify and be in possession of lifelong learning capability.

PROJECT GUIDELINES

Preamble: Project work has been made a part of BCA course to give students exposure to Software development exercises. The primary emphasis of the project work is to understand and gain the knowledge of the principles of software engineering practices. As such, during the development of the project students shall involve themselves in all the stages of the software development life cycle (SDLC) like requirements analysis, systems design, software development/coding, testing and documentation, with an overall emphasis on the development of reliable software systems. Since, the project work spans over the entire final semester, the students shall be advised to take up projects for solving problems of software industry or any research organization or the real life problems suggested by the faculty in-charge of BCA project work in the Institutions. Topic chosen of work must be nontrivial, analytical and application-oriented. It must involve substantial original work and/or development effort based on the theme. Solved, off-the- shelf and pirated work is not entertained. Any attempt of plagiarism or use of unfair means will result in rejection of the work. All activities of the Project Development must be time-bound and the equal participation of the team members expected throughout the Development process.

PROJECT VALUATION

External and Internal Examiners together conduct project valuation objectively. To begin with, the finer details about various points contained in the scheme of valuation may be conclusively agreed upon through mutual consultation. During project evaluation, a student shall present his/her work through live demonstration of the software application developed as a part of project. However, if live demonstration is not possible due to the reason that some companies do not divulge source code on account of ownership rights or copyrights, students may be allowed to make PPT presentation of their authentic works. In such cases, candidates shall produce necessary declarations issued by the companies to this effect. However, students shall be enabled to present their work in entirety. The primary objective of project evaluation shall be to assess the

extent of effort that was put in to meet the objectives of the project and also to gauge the understanding gained by the students in course of their project works. While evaluating Project Reports, examiners shall scrutinize whether Software Development Life Cycle (SDLC) principles have been consistently followed in the project work and the same are documented well in the Reports. However, the relative and overall emphasis of these principles to a particular problem domain chosen may be taken into account so that project evaluations remain fair and objective.

FORMAT OF PROJECT SYNOPSIS

Synopsis is a brief outline or general view, as of a subject or written work; an abstract or a summary of the Project Work. It must be as brief (NOT MORE THAN 20 A4 sized paper pages) as is sufficient to explain the objective and implementation of the project that the candidate is going to take up.

The write up must adhere to the guidelines and should include the following:

1. Title of the Project.
2. Introduction, objectives and scope of the Project.
3. Project Category (Database/Web Application/ Client-server/Networking/ Multimedia/gaming etc.).
4. Tools / Platform, Hardware and Software Requirement specifications.
5. Analysis (DFDs at least up to second level, ER Diagrams/ Class Diagrams, Database Design etc. as perthe project requirements).
6. A complete structure which includes: Number of modules and their description to provide an estimation of the student's effort on the project, Data Structures as per the project requirements for all the modules, Process logic of each module, testing process to be used, reports generation (Mention tentative content of report).
7. Whether Industry Defined/Client Defined/User Defined Project? Mention the type. Mention the Name and Address of the Industry/Client.
8. Limitation of the project.

9. Future scope and further enhancement of the project.

GUIDELINES FOR PREPARATION OF DISSERTATION

1. ORGANISATION OF THE DISSERTATION

The dissertation shall be presented in a number of chapters, starting with Introduction and ending with Conclusion. Each of the chapters will have precise title reflecting the contents of the chapter. A chapter can be subdivided into sections, sub- sections and sub-sub-section so as to present the content discretely and with due emphasis.

Sequence of items in Dissertation Report

The following sequence may be followed in the preparation of the final dissertation report:

- Cover Page (On the hardbound cover)
- Title Page (Inner Cover Page)
- Certificate from the Institute
- Certificate from the Company
- Declaration
- Acknowledgement
- (Detailed) Table of Contents (with page numbers).
- List of Figures (with figure number, figure titles and page numbers)

- List of Tables with table number, table title and page number.

Chapters

1. Introduction

i. Introduction of the System

a. Project Title

b. Category

c. Overview

ii. Background

a. Introduction of the Company

b. Brief note on Existing System

iii. Objectives of the System

iv. Scope of the System

v. Structure of the System

vi. System Architecture

vii. End Users

viii. Software/Hardware used for the development

ix. Software/Hardware required for the implementation

2. SRS

i. Introduction (Brief write-up about SRS)

ii. Overall Description

a. Product perspective

b. Product Functions

c. User characteristics

d. General constraints

e. Assumptions

iii. Special Requirements (Software / Hardware - if any)

iv. Functional requirements

a. Module 1

b. Module 2

c. ….

d. Design Constraints

e. System Attributes

f. Other Requirements (if any)

3. System Design (Functional Design)

i. Introduction (brief write-up about System Design)

ii. Assumptions and Constraints

iii. Functional decomposition

a. System software architecture

b. System technical architecture

c. System hardware architecture

d. External interfaces (if any)

iv. Description of Programs

a. Context Flow Diagram (CFD)

b. Data Flow Diagrams (DFDs – Level 0, Level 1, Level 2)

v. Description of components

a. Functional component 1

b. Functional component 2

c….

4. Database Design (or Data structure)

i. Introduction (brief write-up about Database design)

ii. Purpose and scope

iii. Database Identification

iv. Schema information

v. Table Definition

vi. Physical design

vii. Data Dictionary

viii. ER diagram

ix. Database Administration

a. System information

b. DBMS configuration

c. Support software required

d. Storage requirements

e. Backup and recovery

5. Detailed Design (Logic design of modules)

i. Introduction (brief write-up about Database design)

ii. Structure of the software package (structure chart)

iii. Modular decomposition of the System

a. Module1

a. Inputs

b. Procedural details

c. File I/O interfaces

d. Outputs

e. Implementation aspects (if any)

b. Module 2

6. Program code listing

i. Database connection

ii. Authorization / Authentication

iii. Data store / retrieval / update

iv. Data validation

v. Search

vi. Named procedures / functions

vii. Interfacing with external devices (if any)

viii. Passing of parameters

ix. Backup/recovery

x. Internal documentation

xi. ……

7. User Interface (Screens and Reports)

i. Login

ii. Main Screen / Home page

iii. Menu

iv. Data store / retrieval / update

v. Validation

vi. View

vii. On screen reports

viii. Data Reports

ix. Alerts

x. Error messages

xi. ……..

8. Testing

i. Introduction (brief write-up about Software Testing)

ii. Test Reports

a. Unit Testing

b. Integrate Testing

c. System Testing

- Conclusion
- Limitations
- Scope for enhancement (future scope)
- Abbreviations and Acronyms (list)

- Bibliography / References (list in specified format)

Do not include any header or footer in any page of the report. Only page numbers should be mentioned at the bottom center of each page. 'n' copies of dissertation along with soft copy in CD should be preparedby the candidate.

2. DISSERTATION FORMAT

Paper Quality

The dissertation shall be printed on white bond paper, whiteness 95% or above, weight 70 gram or more per square meter.

Size

The size of the paper shall be standard A4; height 297 mm, width 210 mm.

Type-Setting, Text Processing and Printing

The text shall be printed employing LaserJet or Inkjet printer, the text having been processed using a standard text processor. The standard font shall be Times New Roman of 12 pts with 1.5 line spacing.

Page Format

The printed sheets shall have the following writing area and margins:Top margin .5"

Bottom margin .5"

Left margin 1"

Right margin .75"

Pagination

Page numbering in the text of the dissertation shall be numerals starting from **'1' at the center of the footer**. The text of the written dissertation shall not be less than 60 pages excluding references, tables, questionnaires and other annexure**. Pagination**

for pages before the Introduction chapter shall be in lower case Roman numerals,

e.g., 'iv'.

Paragraph format

Vertical space between paragraphs shall be about 2.5 line spacing.

Five characters or 12 mm should normally indent the first line of each paragraph. A candidate may, however, choose not to indent if (s) he has provided sufficient paragraph separation.

A paragraph should normally comprise more than one line. A single line of a paragraph shall not be left at the top or bottom of a page (that is, no windows or orphans should be left).

The word at the right end of the first line of a page or paragraph should, as far as possible, not be hyphenated.

Chapter and Section format

Chapter

Each chapter shall begin on a fresh page with an additional top margin of about 75 mm. Chapter number (in Hindu- Arabic) and title shall be printed at the center of the line in 6 mm font size (18 pt) in bold face using both upper and lower case (all capitals or small capitals shall not be used). A vertical gap of about 25 mm shall be left between the chapter number and chapter title lines and between chapter title line and the first paragraph.

Sections and Sub- sections

A chapter can be divided into **Sections, Sub-sections and Sub-sub-sections** to present different concepts separately. Sections and sub-sections can be numbered using decimal points, e.g., 2.2 for the second Section in Chapter 2 and 2.3.4 for the fourth Sub-section in third Section of Chapter 2. Chapters, Sections and Sub-Sections shall be included in the **Contents** with page numbers flushed to the right.

Further subsections need not be numbered or included in the contents. The Sections and Sub-sections titles along with their numbers in 5 and 4mm (16 and 14 pt) fonts, respectively, in bold face shall be flushed to the left (not centered) with 15 mm space above and below these lines. In further subdivisions, character size of 3 and 3.5 with bold face, small caps, all caps and italics may be sued for the titles flushed left or centered. These shall not feature in the contents.

Table / Figure Format

As far as possible tables and figures should be presented in portrait style. Small size table and figures (less than half of writing area of a page) should be incorporated within the text, while larger ones may be presented in separate pages. Table and figures shall be numbered chapter- wise. For example, the fourth figure in Chapter 5 will bear the number **Figure 5.4** or Fig.5.4 ***Table number and title will* be *placed above the table while the figure number and caption will be located below the figure***. Reference for Table and Figures reproduced from elsewhere shall be cited in the last and separate line in the table and figure caption,

e.g. (after McGregor [12]).

AUXILIARY FORMAT

Binding

The dissertation shall be hard cover bound in leather or rexin.

Front Covers

The front cover shall contain the following details:

- Full title of dissertation in 6 mm 22-point size font properly centered and positioned at the top.
- Full name of the candidate in 4.5 mm 15-point size font properly centered at the middle of the page.
- A 40 mm dia replica of the college emblem followed by the name of the Department and the yearof submission, each in a separate line and properly centered

and located at the bottom of the page.

Lettering

All lettering shall be embossed in gold.

Bound back

The degree, the name of the candidate and the year of submission shall also be embossed on the bound (side) in gold.

Blank sheets

In addition to the white sheets (binding requirement) two white shall be put at the beginning and end of thedissertation.

Title sheet

This shall be the first printed page of the dissertation and shall contain the submission statement: the Dissertation submitted in partial fulfillment of the requirements of the BCA, the name and Roll No. Of the candidate, name (s) of the supervisor and co-supervisor (s) (if any), Department and year of submission