SRI DHARMASTHALA MANJUNATHESHWARA COLLEGE, UJIRE-574240

(Autonomous)

(Re-Accredited by NAAC at 'A' Grade with CGPA 3.61 out of 4)

DEPARTMENT OF

COMPUTET SCIENCE

Syllabus of

BACHELOR OF VOCATIONAL

[B VOC] PROGRAMME

ON

APP. & SOFTWEAR DEVELOPMENT

(SEMESTER SYSTEM)

2019- 2020 onwards.

Approved by the BOS meeting held on 04-10-2019

Approved by the Academic Council meeting, held on 10-10-2019

Bachelor of Programme

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 Application and Software Development

Introduction to App & Software 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.

Theory Internal Assessment Pattern for Application and Software Development.

THEORY INTERNAL ASSESSMENT PATTERN			
SEMESTER INTERNALS	TIME	MARKS	MODE OF CONDUCT

INTERNAL-I	1 HOUR	30	OFFLINE
INTERNAL-II	1 HOUR	30	ONLINE / MCQ
TOTAL		60	60 Marks converted to 15 and +5(Assignment 2.5 and Attendance 2.5)
FINAL TOTAL		20 MARKS	

Practical Internal and External Assessment Pattern for Application and Software Development.

PRACTICAL INTERNAL ASSESSMENT PATTERN			
SEMESTER INTERNALS	TIME	MARKS	MODE OF CONDUCT
INTERNAL-I	1 HOUR	30	20 WRITTEN + 5 ASSIGNMENT +5 RECORD /ATTENDANCE
INTERNAL-II	2 HOURS	50	30 WRITTEN + 10 ASSIGNMENT +10 RECORD /ATTENDANCE
TOTAL		80	80 Converted into 20 and +10 for Assignment/Record/Attendance
FINAL TOTAL		30 MARKS	

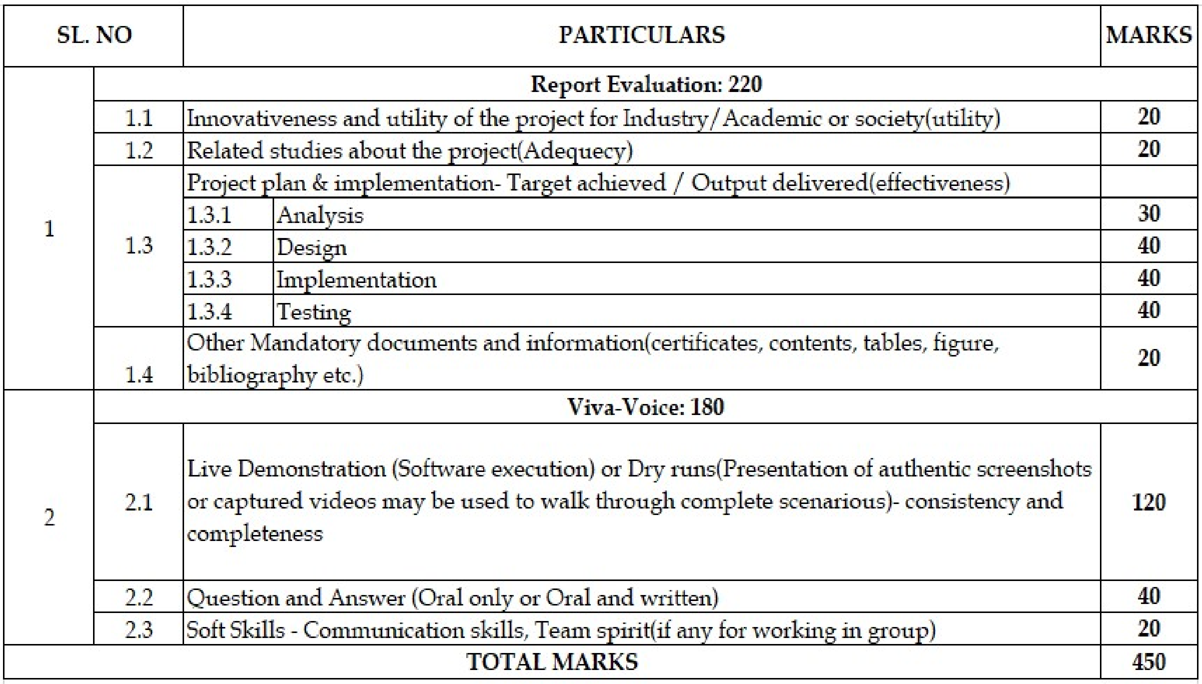
PRACTICAL EXTERNAL ASSESSMENT PATTERN			
SEMESTER END EXAM	TIME	MARKS	MODE OF CONDUCT
EXTERNAL EXAM	3 HOURS	120	Write Up: PART-A(20) +PART-B(40) = 60 Part-A: 1 Question(20Marks) Part-B: 2 Questions(20+20 Marks) Execution: PART-A(10) + PART-B(20) = 30 Part-A: 1 Question(10 Marks) Part-B: 2 Questions(10+10 Marks)

			Viva + Record: 10 + 20 = 30

Final Exam Question Paper Pattern

FINAL EXAM QUESTION PAPER PATTERN			
OPTIONS	NUMBER OF QUESTIONS IN EACH PART	HEADING OF EACH PART	MARKS
PART -A	12	Answer any TEN Questions from the followingeach carries 2 Marks (2*10=20)	2 * 10 = 20
PART-B Unit-I (2 Question) Unit-II (2 Question) Unit-III (2 Question) Unit-IV (2 Question)	8	Answer any One full question from each unit. Example: Unit-I (5 +5 +5) 1.a, 1.b, 1.c OR 2.a, 2.b, 2.c	4 * 15 = 60
TOTAL			80

Project Evaluation Scheme



SL. NO			PARTICULARS	MARKS
1	Report Evaluation: 220			
	1.1	Innovativeness and utility of the project for Industry/Academic or society(utility)		20
	1.2	Related studies about the project(Adequecy)		20
	1.3	Project plan & implementation- Target achieved / Output delivered(effectiveness)		
		1.3.1	Analysis	30
		1.3.2	Design	40
		1.3.3	Implementation	40
		1.3.4	Testing	40
	1.4	Other Mandatory documents and information(certificates, contents, tables, figure, bibliography etc.)		20
2	Viva-Voice: 180			
	2.1	Live Demonstration (Software execution) or Dry runs(Presentation of authentic screenshots or captured videos may be used to walk through complete scenarious)- consistency and completeness		120
	2.2	Question and Answer (Oral only or Oral and written)		40
	2.3	Soft Skills - Communication skills, Team spirit(if any for working in group)		20
TOTAL MARKS				450

I-SEMESTER

Subject Code	Subject Title	Marks		
		IA	Term End Exam	Total
BVOCKAN101	Kannada	20	80	100
BVOCENG101	English	20	80	100
BVOCEF101	Elective	10	40	50
BVOCAS101	Introduction to Python Programming	20	80	100
BVOCAS102	Internet Programming	20	80	100
BVOCASP103	Introduction to Python Programming Practical	30	120	150
BVOCASP104	Internet Programming Practical	30	120	150
BVOCASP105	Project-IBasic Computer Skills Project	30	120	150
Grand Total		**180**	**720**	**900**

II-SEMESTER

Subject Code	Subject Title	Marks		
		IA	Term End Exam	Total
BVOCKAN151	Kannada	20	80	100
BVOCENG151	English	20	80	100
BVOCEF151	Elective	10	40	50
BVOCAS151	Database Management System	20	80	100
BVOCAS152	Introduction to Web Programming	20	80	100
BVOCASP153	DBMS-Lab	30	120	150
BVOCASP154	Web Programming Lab	30	120	150
BVOCASP155	Project-II - Computer Programming Project	30	120	150
Grand Total		**180**	**720**	**900**

III-SEMESTER

Subject Code	Subject Title	Marks		
		IA	Term End Exam	Total
BVOCKAN201	Kannada	20	80	100
BVOCENG201	English	20	80	100
BVOCEF201	Elective	10	40	50
BVOCAS201	Python for Data Science	20	80	100
BVOCAS202	Object Oriented Programming with Java	20	80	100
BVOCASP203	Python for Data Science-Lab	30	120	150
BVOCASP204	Java Programming -Lab	30	120	150
BVOCASP205	Project-III - Kotlin Programming Project	30	120	150
Grand Total		**180**	**720**	**900**

IV-SEMESTER

Subject Code	Subject Title	Marks		
		IA	Term End Exam	Total
BVOCKAN251	Kannada	20	80	100
BVOCENG251	English	20	80	100
BVOCEF251	Elective	10	40	50
BVOCAS251	Mobile Application Development	20	80	100
BVOCAS252	Introduction To Go Programming	20	80	100
BVOCASP253	Mobile Application Development Lab	30	120	150
BVOCASP254	Introduction To Go Programming-Lab	30	120	150
BVOCASP255	Project-IV - Data Structure and C++ Project	30	120	150
Grand Total		**180**	**720**	**900**

V-SEMESTER

Subject Code	Subject Title	Marks		
		IA	Term End Exam	Total
BVOCAS301	ASP .NET CORE Programming	20	80	100
BVOCAS302	Machine Learning With Python	20	80	100
BVOCAS303	Unix And Shell Programming	20	80	100
BVOCAS304	DevOps For Developers	20	80	100
BVOCASP305	Machine Learning With Python-Lab	30	120	150
BVOCASP306	Unix And Shell Programming-Lab	30	120	150
BVOCASP307	Project-V - DevOps For Developers -Project	30	120	150
Grand Total		**170**	**680**	**850**

VI-SEMESTER

Subject Code	Subject Title	Marks		
		IA	Term End Exam	Total
BVOCAS351	Internet Of Things	20	80	100
BVOCAS352	Computer Networks	20	80	100
BVOCAS353	Ethical Hacking Techniques	20	80	100
BVOCAS354	NoSQL Database	20	80	100
BVOCASP355	Project Work	50	400	450
Grand Total		**130**	**720**	**850**

INTRODUCTION TO PYTHON PROGRAMMING SEMESTER – I			
Subject Code	**BVOCAS101**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03
CREDITS – 03			

Course objectives: This course will enable students to

CO1: Learn Syntax and Semantics and create Functions in Python.

CO2: Handle Strings and Files in Python.

CO3: Understand Lists, Dictionaries and Regular expressions in Python.

CO4: Implement Object Oriented Programming concepts in Python.

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

Library Modules.	
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**
Course outcomes: The students should be able to:	
CO1: Examine python syntax and semantics and be fluent in the use of python flow control and functions. CO2: Demonstrate proficiency in handling strings and file systems. CO3: Create, run and manipulate python programs using core data structures like lists, dictionaries and use regular expressions. CO4: Interpret the concepts of object-oriented programming as used inpython CO5: Implement exemplary applications related to network programming, web services and databases inpython	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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 • 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 University Press. 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 PROGRAMMING

SEMESTER – I

Subject Code	**BVOCAS102**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Illustrate the Semantic Structure of HTML and CSS

CO2: Compose forms and tables using HTML and CSS

CO3: Examine JavaScript framework.

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; Basic Syntax; Standard XHTML Document Structure; Basic Text Markup.	**9Hours**
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, CSS Drop Down, JavaScript: Client Side Scripting, What is JavaScript and What can it do?,	**9 Hours**

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 Date get and set methods, JS Loop For, JS Loop While

Course outcomes: The students should be able to:

CO1: Adapt html and css syntax and semantics to build webpages.

CO2: Construct and visually format tables and forms using html andcss

CO3:Developclient-sidescriptsusingJavaScriptandserver-sidescriptsusingphptogenerateand display 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 on core features

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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
- 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.
2. 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 – 06

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

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 their total marks. Find out topper and his/her score.

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

Conduction of Practical Examination:

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

Change of experiment is allowed only once and marks allotted to the procedure part to be 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 – 06

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 user clicks 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 are allowed to pick three experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

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

BASIC COMPUTER SKILLSPROJECT

SEMESTER – 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 – 06

Course objectives: This course will enable students to

CO1: Will learn and understand the Basics of Computers and apply the application tools like word processor, spread sheet and 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 physical components 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 class time 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 and Output 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 different Models of Computers. Use Presentation tool.

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 internetoperation

CO2: Demonstrate skills using wordprocessor

CO3: Demonstrate skills using spreadsheetpresentation

CO4: Demonstrate skills usingpresentation

Conduction of Practical Examination:

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

Change of experiment is allowed only once and marks allotted to the procedure part to be 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	**BVOCAS151**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: To study the concepts of web application development such as XHTML, XML, PHP, Java web software, and Database access through JDBC and PHP.

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 and XPointer, XML Validator, XML DTD, XML Schema, XML on the Server, AJAX Introduction.	**9Hours**
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	**9 Hours**

Clause, PHP MySQL Use The ORDER BY Clause, PHP MySQL Update Data.

Course outcomes: The students should be able to:

CO1: Discuss the fundamentals of web and concept ofxhtml.

CO2: Describe different concepts of JavaScript and xhtml documents and construct dynamic documents with Java Script.

CO3: Describe xml using the user defined tags, dtd, namespaces and schemas with simple programs

CO4: Discuss the concepts of php with associate dprograms

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

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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
- 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 SYSTEMS

SEMESTER – II

Subject Code	**BVOCAS152**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Provide a strong foundation in database concepts, technology, and practice.

CO2: Practice SQL programming through a variety of database problems.

CO3: Demonstrate the use of concurrency and transactions in database

CO4: Design and build database applications for real world problems.

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, 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.	**9 Hours**
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**
Course outcomes: The students should be able to:	
CO1: Identify, analyze and define database objects, enforce integrity constraints on a database using rdbms. CO2: Use structured query language (sql) for database manipulation. CO3: Design and build simple database systems develop application to interact with databases	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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 • The students will have to answer 4 full questions, selecting one full question from each module	
Text Books:	
1.Fundamentals of Database Systems, Sixthedition, 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 Ramakrishnan 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 PROGRAMMINGLABORATORY

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 – 06

Course objectives: This course will enable students to

CO1: To study the concepts of web application development such as XHTML, XML, PHP, Java web software, and Database access through JDBC and PHP.

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 tableformat.
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 for the following problems:

a) Parameter: Astring

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 amatrix.

c) Multiplication of two matrices.

d) Addition of two matrices

8. Write a PHP program named states.py that declares variable states with value “Mississippi Alabama 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 named states 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 the list.

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.

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

Conduction of Practical Examination:

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

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

DATABASE MANAGEMENT SYSTEMS LABORATORY

SEMESTER – 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 – 06

Course objectives: This course will enable students to

CO1: Foundation knowledge in database concepts, technology and practice to groom students into well-informed database application developers.

CO2: Strong practice in SQL programming through a variety of database problems.

CO3: Develop database applications using front-end tools and back-end DBMS

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 database constraints.

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 on Android/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 copies in each branch, etc.

2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2019 to Jun 2020
3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
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 (Use UNION 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 must also 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 JOIN operation).

4. Find the title of movies and number of stars for each movie that has at least one rating and find the 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 for all 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 10 percent raise.

3. Find the sum of the salaries of all employees of the ‘Accounts’ department, as well as the maximum 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.

Course outcomes: The students should be able to:

CO1: Create, update and query on thedatabase.

CO2: Demonstrate the working of different concepts ofdbms

CO3: Implement, analyze and evaluate the project developed for anapplication

Conduction of Practical Examination:

1. All laboratory experiments from part A are to be included for practical examination.
2. Mini project has to be evaluated 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 is allowed only once and marks allotted to the procedure part to be made zero.

COMPUTER PROGRAMMING PROJECT

SEMESTER – 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 – 06

Course objectives: This course will enable students to

CO1: Write flowcharts, algorithms and programs.

CO2: Familiarize the processes of debugging and execution.

CO3: Implement basics of C programming language.

CO4: Illustrate solutions to the laboratory programs.

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 and use 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 units 80 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 read the 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: Don’t use library function sqrt(n).
13. Implement structures to read, write and compute average marks and the students scoring above and 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

Course outcomes: The students should be able to:

CO1: Write algorithms, flowcharts and program for simpleproblems.

CO2: Correct syntax and logical errors to execute aprogram.

CO3: Write iterative and wherever possible recursiveprograms

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

Conduction of Practical Examination:

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

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

PYTHON FOR DATA SCIENCE

SEMESTER – III

Subject Code	**BVOCAS201**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Explore Python language fundamentals, including basic syntax, variables, and types

CO2: Create and manipulate regular Python lists

CO3: Use functions and import packages

CO4: Build Numpy arrays, and perform interesting calculations

CO5: Create and customize plots on real data

CO6: 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 pythonscripts.

CO2: Sequence and file operations inpython.

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

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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
- 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 practical approach , by E R Rajiv Chopra, S Chand publications.

OBJECT ORIENTED PROGRAMMING WITH JAVA

SEMESTER – III

Subject Code	**BVOCAS202**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

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.

Module – 1	**Teaching Hours**
Java introduction, java syntax, java comments, java variables, java datatypes, java type casting, java operators, java string, java match, java Booleans, java if else, java switch, java while loop, java for loop, java break/continue, java array.	**9 Hours**
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, java wrapper 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**

Course outcomes: The students should be able to:

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

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

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 computerlanguage

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

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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
- 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, Pearson Education,2008, ISBN:9788131720806.
2. Rajkumar Buyya,S Thamarasi selvi, xingchen chu, Object oriented Programming with java, Tata McGraw 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 – 06

Course objectives:

CO1: Gain knowledge about basic Java language syntax and semantics to write Java programs and use

CO2: Concepts such as variables, conditional and iterative execution methods etc.

CO3: Understand the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc. and exception handling mechanisms.

CO4: Understand the principles of inheritance, packages and interfaces.

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.

Course outcomes: The students should be able to:

CO1: Identifyclasses,objects,membersofaclassandrelationshipsamongthemneededforaspecific problem

CO2: Write java application programs using oop principles and proper programstructuring

CO3: Demonstrate the concepts of polymorphism andinheritance

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

CO5: Understand and apply object oriented features and javaconcepts

Conduction of Practical Examination:

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

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

PYTHON FOR DATA SCIENCE LABORATORY

SEMESTER – 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 – 06

Course objectives: This course will enable students to

CO1: Explore Python language fundamentals, including basic syntax, variables, and types

CO2: Create and manipulate regular Python lists

CO3: Use functions and import packages

CO4: Build Numpy arrays, and perform interesting calculations

CO5: Create and customize plots on real data

CO6: Supercharge your scripts with control flow, and get to know the Pandas Data Frame.

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) stored about 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. Raise an 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 print it. 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 between 1 to 100

b) Create a new column such that, each row contains the row number of nearest row-record by Euclidean 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 having family 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 the order 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 so that 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)

Course outcomes: The students should be able to:

CO1: Advanced concepts of python like writing pythonscripts.

CO2: Sequence and file operations inpython.

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

Conduction of Practical Examination:

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

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

KOTLIN PROGRAMMING PROJECT

SEMESTER – III

Laboratory Code	**BVOCASP205**	**IA Marks**	30
Number of Lecture Hours/Week	06	**Exam Marks**	120
Total Number of Lecture Hours	72	**Exam Hours**	03

CREDITS – 06

Course objectives:

CO1: Create professional applications using Kotlin, the new Java-based programming language

CO2: Understand the basics of object-oriented software development, the most important development paradigm

CO3: Use Intellij, the popular Java (and Kotlin) IDE, to write code effectively and professionally

CO4: Understand the concepts of the Kotlin language and how it integrates neatly with Java

CO5: Understand the principles behind other object-oriented languages (like Java, C++, PHP, C#, Scala, or Swift)

Note: Carry out a Mini Project which includes two different applications with the help of open source tools by studying the below modules.

Section-A

Module1: Get started

You learn how to work with the Kotlin REPL (Read-Eval-Print Loop) interactive shell, and you practice using the basic syntax of Kotlin code.

Module2: Kotlin basics

You learn how to use Kotlin data types, operators, and variables, and how to work with booleans and conditions. You explore the difference between null able and non-null able variables, and you practice using arrays, lists, and loops in Kotlin.

Module 3: Functions

You learn how to create a program with a main () function and arguments in IntelliJ IDEA. You create small programs as you learn about default values, compact functions, list filters, basic lambdas, and higher-order functions.

Module 4: Classes and objects

You learn about classes, objects, and inheritance in Kotlin. You create small programs as you learn about abstract classes, interfaces, and interface delegation.

Module 5.1: Extensions

You learn about collections, constants, and extension functions in Kotlin. You create small programs as you learn about pairs, triples, lists, and hash maps for storing data, and implement extension functions to add functionality to existing classes.

Module 5.2: Generics

You learn about generic classes, methods, and functions in Kotlin. You create a type hierarchy, make classes more flexible by making them generic, and extend their functionality with generic methods and functions.

Module 6: Functional manipulation

You learn about annotations, labeled breaks, and Single Abstract Methods (SAMs). You also review lambdas and higher-order functions.

You then create and use lambdas and higher-order functions, and learn about higher-order functions in the Kotlin Standard Library.

Course outcomes: The students should be able to:

CO1: Develop skills to use android studio to buildapps.

CO2: Run app on a device or in theemulator

CO3: Be skilled to add interactivebuttons.

CO4: Build app functionality andui

CO5: Tackle databases and api

CO6: Perform unittesting

CO7: Fixbugs

Text Books

1. Kotlin for Android Developers Learn Kotlin the easy way while developing an Android App
   © 2015 - 2016 Antonio Leiva
2. Programming Kotlin Familiarize yourself with all of Kotlin's features with this in depth guide by Stephen Samuel and Stefan Bocutiu
3. Kotlin in Action by Dmitry Jemerov and Svetlana Isakova.

Reference Books

1. Android Development with Kotlin, Learn Android application development with the extensive features of Kotlin by Marcin Moskala and Igor Wojda.
2. Kotlin Language Documentation.

References: Online Resources

- https://kotlinlang.org/
- https://codelabs.developers.google.com/
- https://developer.android.com/kotlin/learn
- https://codelabs.developers.google.com/kotlin-bootcamp/
- https://codelabs.developers.google.com/android-kotlin-fundamentals/
- https://codelabs.developers.google.com/advanced-android-kotlin-training/
- https://openclassrooms.com/en/courses/5774406-learn-kotlin/6137246-get-the-most-out-of-this-course
- https://books.goalkicker.com/
- https://www.cosmiclearn.com/kotlin/index.php

Conduction of Practical Examination:

- Students has to carry out a live demonstration of two different Mobile Application
- Each demonstration of carries 45 Marks each
- Viva-Voice carries 10 Marks
- Record carries 20 Marks
- Strictly follow the instructions as printed on the cover page of answer script

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

MOBILE APPLICATION DEVELOPMENT

SEMESTER – IV

Subject Code	**BVOCAS251**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Learn to setup Android application development environment

CO2: Illustrate user interfaces for interacting with apps and triggering actions

CO3: Interpret tasks used in handling multiple activities

CO4: Identify options to save persistent application data

CO5: Appraise the role of security and performance in Android applications

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,	**9 Hours**

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.	
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**
Course outcomes: The students should be able to:	
CO1: Create, test and debug android application by setting up android developmentenvironment CO2: Implement adaptive, responsive user interfaces that work across a wide range ofdevices. CO3: Infer long running tasks and background work in androidapplications CO4: Demonstrate methods in storing, sharing and retrieving data in androidapplications CO5: Analyzeperformanceofandroidapplicationsandunderstandtheroleofpermissionsandsecurity CO6: Describe the steps involved in publishing android application to share with theworld	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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 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 India Pvt 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’Reilly SPD Publishers, 2015.
3. Anubhav Pradhan, Anil V Deshpande, “ Composing Mobile Apps” using Android, Wiley 2014, ISBN: 978-81-265-4660-2

INTRODUCTION TOGO PROGRAMMING

SEMESTER – IV

Subject Code	**BVOCAS252**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Learn about what makes Go a great language

CO2: Learn how to install the Go tool chain

CO3: Learn how to setup Visual Studio Code to edit and debug Go programs

CO4: Learn how to work with the Go Playground to test and run snippets of Go code

CO5: Learn and understand the basic Go language syntax and features

CO6: Learn how to use the Go tool chain commands to compile, test, and manage Go code

CO7: And finally, you'll learn how to work with and manage Go modules for module dependency management

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. Value, Example: DecodingS-E xpressions, Accessing Struct Field Tags, Displaying the Methods of a Type, A Word of Caution	**9 Hours**

Course outcomes: The students should be able to:

CO1: Learn go fundamentals and apply them in real worldscenarios

CO2: Get to grip with advanced features like go language andconcurrency

CO3: Understand and develop your knowledge of programmingfundamentals

CO4: Build up a knowledge foundation for more advanced programminglanguages

CO5: Learn the language behind the super popular Dockertechnology

CO6: Everything you need to get up andgo

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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

  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 – 06

Course objectives: This course will enable students to

CO1: Describe those aspects of mobile programming that make it unique from programming for other platforms,

CO2: Critique mobile applications on their design pros and cons,

CO3: Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces,

CO4: Program mobile applications for the Android operating system that use basic and advanced phone features.

CO5: Deploy applications to the Android marketplace for distribution.

Laboratory Experiments:

1. To develop an application that uses GUI Components, Fonts and Colors.
2. To develop an application that uses Layout Managers and Event Listeners.
3. To develop an active calculatorapplication.
4. To developanapplicationthatmakesuseofdatabase.
5. To developanactiveapplicationthatusesGPSlocationinformation.
6. To implementanapplicationthatwritesdatatotheSDcard.
7. To developanapplicationthatdrawsbasicgraphicalprimitivesonthescreen.
8. To developanapplicationthatmakesuseofRSSFeed.
9. To implementanapplicationthatimplementsmultithreading.
10. To implementanapplicationthatcreatesanalert uponreceivingamessage.
11. To implementanapplicationthatcreatesalarmclock.

Course outcomes: The students should be able to

CO1: Apply essential android programmingconcepts.

CO2: Develop various android applications related to layouts & rich uses interactiveinterfaces

CO3: Gain knowledge concerning mobile operating systems and theirarchitecture

CO4: Recognize and setup a mobile device and application runtimeenvironment

CO5: Be able to setup programming tools for a mobile application developer (forselected modern mobileplatforms)

CO6: Understands the need for continuous improvement of his/her skills due to therapidly changing environment of mobiledevices

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 are allowed to pick three experiments from the lot.
- Strictly follow the instructions as printed on the cover page of answer script

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

INTRODUCTION TOGO 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 – 06			

Course objectives: This course will enable students to

CO1: Master the fundamental syntax of Go, including variable declarations, data types control structures, and functions

CO1: Test, build, and deploy command-line Go applications

CO1: Benchmark your Go applications

CO1: Master use of the Go documentation

CO1: Manage state

CO1: Build web services

CO1: Write concurrent code

CO1: Produce code more efficiently with templating

CO1: Package and deploy complete Go applications

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

Course outcomes: The students should be able to

CO1: Understand the fundamentals of go programminglanguage.

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

CO3: Boost your hireability through innovative and independentlearning.

CO4: Understand and develop your knowledge of programmingfundamentals

CO5: Build up a knowledge foundation for more advanced programminglanguages

Conduction of Practical Examination:

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

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

DATA STRUCTURES AND APPLICATIONS PROJECT

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 – 06

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

CO1: Explain fundamentals of data structures and their applications essential for programming/problem solving.

CO2: Illustrate linear representation of data structures: Stack, Queues, Lists, Trees and Graphs.

CO3: Demonstrate sorting and searching algorithms.

CO4: Find suitable data structure during application development/Problem Solving.

CO5: Asymptotic performance of algorithms.

CO6: Linear data structures and their applications such as stacks, queues and lists

CO7: Non-Linear data structures and their applications such as trees and graphs

CO8: Sorting and searching algorithms

NOTE: Students has to go with self-learning with the below theory topics mentioned in the Section-A, and Carry out experiments mentioned in the Section-B.

Section-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

Section-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 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 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 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

7. 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

8. 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 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 Circular Linked 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 result in 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 on Binary 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) of Cities

    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/BFS method

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

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 their applications

CO7: Implement, analyze and evaluate the searching and sorting algorithms

CO8: Choose the appropriate data structure for solving real world problems

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

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.

Conduction of Practical Examination:

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

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

ASP .NET CORE PROGRAMMING

SEMESTER – V

Subject Code	**BVOCAS301**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Understand the goals and benefits of the .NET Core platform

CO2: Learn how to build a compelling and maintainable HTML user interface using the Razor view engine and client-side JavaScript

CO2: Client side programming: HTTP, CGI, Cookies, JavaScript, HTML, XML.

CO3: ASP.NET Web services and web service security.

CO4: Introduction to the .NET framework.

CO5: NET Interoperation services.

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, Multiple Inheritances, 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	**9 Hours**

Exceptions,Delegates	
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**
Course outcomes: The students should be able to:	
CO1: Learn about ms.net framework developed byMicrosoft. CO2: Be able to using xml in c#.net specifically ado.net and sqlserver CO3: Be able to understand use of c# basics, objects and types,inheritance CO4: Develop, implement and creating applications withc#. CO5:Develop, implement, and demonstrate component services, threading, remoting,windows services, web CO6: Understandandbeabletoexplainsecurityinthe.netframeworkanddeploymentinthe net.	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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 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. (For Programming 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 – HerbertSchildt, 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/

MACHINE LEARNING WITH PYTHON

SEMESTER – V

Subject Code	**BVOCAS302**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Conceptualization and summarization of big data and machine learning,

CO2: Trivial data versus big data,

CO3: Big data computing technologies,

CO4: Machine learning techniques, and scaling up machine learning approaches.

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 for Machine Learning. **Clustering:** Finding similarity distance, K-Means Clustering, Creating Product Segments using Clustering, Hierarchical Clustering.	**9 Hours**

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 data for variousapplications.

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

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 scalabilityissues.

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

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

CO6: Be abletointegratemachinelearninglibrariesandmathematicalandstatisticaltoolswithmodern technologies like Hadoop and mapreduce

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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

  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, 2nd edition, springer series in statistics.
2. Ethem Alpaydın, Introduction to machine learning, second edition, MIT press.

UNIX AND SHELL PROGRAMMING

SEMESTER – V

Subject Code	**BVOCAS303**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1:To provide introduction to UNIX Operating System and its File System

CO2:To gain an understanding of important aspects related to the SHELL and the process

CO3: Demonstrate the working of basic commands of Unix environment including file processing

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

CO5: To provide a comprehensive introduction to SHELL programming, services and utilities.

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**

Course outcomes: The students should be able to:

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

CO2: Demonstrate Unix commands for file handling and processcontrol

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 shell script to solve theproblem

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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

  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 LABORATORY

SEMESTER – 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 – 06

Course objectives: This course will enable students to

CO1: The objectives of the course "**Machine Learning and Data Mining**" is to introduce students to state-of-the-art methods and modern programming tools for data analysis.

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 and the 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 consistent with the training examples.
6. Design a program to implement the **Simple Linear Regression Model** for a sample training data set 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

Course outcomes: The students should be able to:

CO1: Understand complexity of machine learning algorithms and theirlimitations;

CO2: Understand modern notions in data analysis orientedcomputing

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

CO4: Be capable of performing distributedcomputations;

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

Conduction of Practical Examination:

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

Change of experiment is allowed only once and marks allotted to the procedure part to be 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 – 06

Course objectives: This course will enable students to

CO1: 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.

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 to the 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 create sub-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 output each filename followed by its permissions.

2b. Write a shell script which accepts valid log-in names as arguments and prints their corresponding home 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 need not include error checking.

5a. Write a shell script that accepts filename as argument and display its creation time if file exist and 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 in the 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. (Use expr command to check the length).

7a. Write a shell script that gets executed and displays the message either “Good Morning” or “Good Afternoon” 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 occurrence of 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 system and 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 period of time.

9a. Write a shell script that accepts the filename, starting and ending line number as an argument and display 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 characters must 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 must remain unchanged.

11a. Write an awkscript to find out total number of books sold in each discipline as well as total book sold using associate array down table as given below.

Electrical-34

Mechanical-67

Electrical-80

Computer Science-43

Civil-98

Mechanical-65

Computer Science-64

11b. Write an awkscript 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.

Course outcomes: The students should be able to:

CO1: Demonstrate the working of basic commands of unix environment including fileprocessing

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

CO3: Implement unix commands/ system calls to demonstrate processmanagement

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 givenproblem

Conduction of Practical Examination:

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

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

DEVOPS FOR DEVELOPERS

SEMESTER – V

Subject Code	**BVOCAS304**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet your business requirements

CO2: Be able to deliver change requests from customers rapidly and effectively by adding new (and updating existing) features

CO3: Obtain complete knowledge of the “version control system” to effectively track changes augmented with Git and Github training

CO4: Have a detailed overview of continuous integration and container ecosystem by learning tools such as Jenkins and Docker

CO5: Successfully transition from a software engineer to a DevOps engineer

Module – 1	**Teaching Hours**
Fundamentals:Beginning DevOps for Developers, Introducing DevOps, Building Blocks of 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**

Course outcomes: The students should be able to:

CO1: Be able to continuous development, continuous testing, and configuration management, including continuous integration and continuous deployment and finally continuous monitoring of the software throughout its devops development lifecycle.

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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

  The 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 PROJECT

SEMESTER – 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 – 06

Course objectives: This course will enable students to

CO1: Understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet your business requirements

CO2: Be able to deliver change requests from customers rapidly and effectively by adding new (and updating existing) features

CO3: Obtain complete knowledge of the "version control system" to effectively track changes augmented with Git and Github training

CO4: Have a detailed overview of continuous integration and container ecosystem by learning tools such as Jenkins and Docker

CO5: Successfully transition from a software engineer to a DevOps engineer

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

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 its DevOPs development lifecycle.

Conduction of Practical Examination:

- Students has to conduct a mini project on laboratory experiments considering twodifferent test cases 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 experiments

  Test Case 1: 45 Marks , Test Case 2: 45 Marks, Viva-Voice: 10 Marks, Record: 20 Marks

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

INTERNET OF THINGS

SEMESTER – VI

Subject Code	**BVOCAS351**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Assess the genesis and impact of IoT applications, architectures in real world.

CO2: Illustrate diverse methods of deploying smart objects and connect them to network.

CO3: Compare different Application protocols for IoT.

CO4: Infer the role of Data Analytics and Security in IoT.

CO5: Identify sensor technologies for sensing real world entities and understand the role of IoT in various domains of 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, The Transport 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**

Course outcomes: The students should be able to:

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

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

CO3: Appraise the role of IoT protocols for efficient networkcommunication.

CO4: Elaborate the need for data analytics and security inIoT

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

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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

  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 Edition VPT, 2014. (ISBN: 978-8173719547)
2. Raj Kamal, “Internet of Things: Architecture and Design Principles”, 1st Edition, McGraw Hill Education, 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 NETWORKS

SEMESTER – VI

Subject Code	**BVOCAS352**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Demonstration of application layer protocols

CO2: Discuss transport layer services and understand UDP and TCP protocols

CO3: Explain routers, IP and Routing Algorithms in network layer

CO4: Disseminate the Wireless and Mobile Networks covering IEEE 802.11 Standard

CO5: Illustrate concepts of Multimedia Networking, Security 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	**9 Hours**

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.	
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-ofService (QoS) Guarantees: Resource Reservation and Call Admission	**9 Hours**
Course outcomes: The students should be able to:	
CO1: Explain principles of application layer protocols CO2: Recognize transport layer services and infer udp and tcpprotocols CO3: Classify routers, ip and routing algorithms in network layer CO4: Understand the wireless and mobile networks covering ieee 802.11standard CO5: Describe multimedia networking and network management	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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 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 TECHNIQUES

SEMESTER – VI

Subject Code	**BVOCAS353**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

CO1: Knowledge about Ethical Hacking.

CO2: Understand the basic concepts of Open Source Intelligence.

CO3: Analyze the Hacking Techniques by developing simple tools

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

Course outcomes: The students should be able to:

CO1: Gain knowledge about ethical hacking and penetrationtesting.

CO2: Learnaboutvarioustypesofattacks,attackersandsecuritythreatsandvulnerabilitiespresent in the computer system.

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

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

CO5:Gainknowledgeofthetools,techniquesandethicalissueslikelytofacethedomainofethical hacking and ethical responsibilities.

Question paper pattern:

- The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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

  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 inComputing, 5th Edition , Pearson Education , 2015

Reference Books:

1. Martti Lehto, Pekka Neittaanmäki, Cyber Security: Analytics, Technology and Automation edited, 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 DATABASE

SEMESTER – VI

Subject Code	**BVOCAS354**	**IA Marks**	20
Number of Lecture Hours/Week	03	**Exam Marks**	80
Total Number of Lecture Hours	36	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students 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 data and 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 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.	**9 Hours**
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	**9 Hours**

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	
Module –4	
Introduction to MongoDB, Installing MongoDB,The Data Model, Working with Data, Advanced Queries, Database Administration, Replication, Sharding	**9 Hours**
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 andgraph). CO2: Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune column-oriented NoSQLdatabases. CO3: Explain the detailed architecture, define objects, load data, query data and performance tune document-oriented NoSQLdatabases	
Question paper pattern:	
• The question paper will have two parts, PART-A (20 Marks) and PART-B (80 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 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 Polyglot Persistence, Pearson Addision Wesley, 2012 2. The Definitive Guide to MongoDB, By Davud Hows, Peter Membrey, Eelco Plugge, Tim Hawkins, 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)

PROJECT WORK

SEMESTER – VI

Subject Code	**BVOCAS355**	**IA Marks**	50
Number of Lecture Hours/Week	Nil	**Exam Marks**	180
Total Number of Lecture Hours	Nil	**Exam Hours**	03

CREDITS – 06

Course Outcomes:

CO1: Have the ability to obtain and use the mathematical, scientific and engineering-based knowledge towards 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 like surroundings, 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 software engineering practices.

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

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

Project Phase-I:

Project Phase-II:

Project Phase-III: Final Evaluation

Scheme of Examinations								
I/II/III/IV Semesters								
	Particulars	**No of courses**	**Instruction Hours/Week**	**Duration of Exam(Hours)**	**Marks**			**Credits**
General Education	2 Languages	2L	2*2	2*3	2*20	2*80	2*100	4
	2 Core Papers	2T	2*3	2*3	2*20	2*80	2*100	6
	1 Elective	1T	1*2	1*2	1*20	1*80	1*100	2
Skill Component	2 Practicals with 1 Project	2P	2*6	2*3	3*20	3*80	3*100	12
		1 Project	1*6					6
V/VI Semesters								
General Education	4 Core Papers	4T	4*3	4*3	4*20	4*80	4*100	12
Skill Component	2 Practicals with 1 project with Internship	2P	2*6	2*3	3*20	3*80	3*100	12
		1 Project	1*6					6
								60