

Academic Council --/--/----
Item No: _____

**SIES College of Arts, Science and Commerce,
Sion (W)
Autonomous College**



**Syllabus for
Program: First Year Bachelor of Science
(NEP) Course: Computer Science
Semester: I & II**

With effect from
Academic Year 2023 -24

Preamble

Information and Communication Technology (ICT) has today become an integral part of all industry domains as well as fields of academics and research. The industry requirements and technologies have been steadily and rapidly advancing. Organizations are increasingly opting for open source systems. The students too these days are thinking beyond careers in the industry and aiming for research opportunities.

The B.Sc. Computer Science course structure therefore needed a fresh outlook and complete overhaul. A real genuine attempt has been made while designing the new syllabus for this 3 year graduate course. Not only does it prepare the students for a career in the Software industry, it also motivates them towards further studies and research opportunities.

The core philosophy of overall syllabus is to -

- a. Form strong foundation of Computer science,
- b. Introduce emerging trends to the students in gradual way,
- c. Groom the students for the challenges of ICT industry

In the first year i.e. for semester I & II, the basic foundation of important skills required for software development is laid. The syllabus proposes to have four core subjects of Computer science and two core courses of Mathematics-Statistics. In Semester II the students would also be given industrial exposure via field projects/industrial visit. All core subjects are proposed to have theory as well as practical tracks. While the Computer Science courses will form fundamental skills for solving computational problems, the Mathematics & Statistics course will inculcate research oriented acumen.

The syllabus design for further semesters encompasses more advanced and specialized courses of Computer Science.

We sincerely believe that any student taking this course will get a very strong foundation and exposure to basics, advanced and emerging trends of the subject. We hope that the students' community and teachers' fraternity will appreciate the treatment given to the courses in the syllabus.

We wholeheartedly thank all experts who shared their valuable feedback and suggestions in order to improvise the contents, we have sincerely attempted to incorporate each of them. We further thank the Chairperson and members of the Board of Studies for their confidence in us. Special thanks to the Department of Computer Science and colleagues from various colleges, who volunteered or have indirectly helped design certain specialized courses and the syllabus as a whole.

Program Outcomes and Program Specific Outcomes

B.Sc. Computer Science

SR.NO	Details
PO 1	Recall and explain acquired scientific knowledge in a comprehensive manner and apply the skills acquired in their chosen discipline. Interpret scientific ideas and relate its interconnectedness to various fields in science.
PO 2	Evaluate scientific ideas critically, analyse problems, explore options for practical demonstrations, illustrate work plans and execute them, organize data and draw inferences.
PO 3	Explore and evaluate digital information and use it for knowledge upgradation. Apply relevant information so gathered for analysis and communication using appropriate digital tools.
PO 4	Ask relevant questions, understand scientific relevance, hypothesize a scientific problem, construct and execute a project plan and analyze results.
PO 5	Take complex challenges; work responsibly and independently, as well as in cohesion with a team for completion of a task. Communicate effectively, convincingly and in an articulate manner.
PO 6	Apply scientific information with sensitivity to values of different cultural groups. Disseminate scientific knowledge effectively for upliftment of the society.
PO 7	Follow ethical practices at the workplace and be unbiased and critical in interpretation of scientific data. Understand the environmental issues and explore sustainable solutions for it.
PO 8	Keep abreast with current scientific developments in the specific discipline and adapt to technological advancements for better application of scientific knowledge as a lifelong learner.

SR.NO	Details
PSO 1	Apply knowledge of computational mathematics, statistics and programming acquired in the field of Computer Science.
PSO 2	Identify, analyze complex problems in the real world and formulate innovative solutions to those problems.
PSO 3	Compare and apply hardware and software technologies for implementing reliable optimized solutions catering to need and available resources.
PSO 4	Apply software development, managerial, Professional, and soft skills in industry
PSO 5	Understand the global needs and prepare themselves for the changing needs worldwide adapting an ability to engage in life-long learning.
PSO 6	Become a responsible, ethical citizen and explore environmental issues to develop sustainable solutions for it.

F.Y.B.Sc. Computer Science Syllabus
Credit Based System and Grading System
Academic year 2023-2024

Semester – I						
Course Code	Course Type	Course Title	Credits	Lectures/Week		
				Theory	Practical (2 lectures)	Total
SIUCSMJ111	Major	Programming with Python	3	3		3
SIUCSMJP111	Major Practical	Practical of SIUCSMJ111	1		1	1
SIUCSMN111	Minor Subject	Fundamentals of Mathematics and Statistics-I	3	3		3
SIUCSMNP111	Minor Practical	Practical of SIUCSMN111	1		1	1
Vocational and Skill Enhancement						
SIUCSVS111	Vocational Skill Course (VSC)	Basic Web Programming	1	1		1
SIUCSVS111	Vocational Skill Course practical	Practical of SIUSCS13	1		1	1
SIUCSSE111	Skill Enhancement Course (SEC)	Beginning MySQL	1	1		1
SIUCSSE111	Skill Enhancement Course (SEC)	Practical of SIUSCSP14	1		1	1
Generic/ Open Elective Courses (OE)						
SIUCSOE111	Open Elective	Basic Web Designing	2	2		2
SIUCSOE111	Open Elective Practical	Practical of SIUSCS15	2		2	2
Ability Enhancement Courses (AEC)						
SIUCSAE111	Ability Enhancement Courses	Professional communication development - I	2	2		2
Value Education Courses (VEC)						
SIUCSVE111	Value Education Courses	Environmental studies	2	2		2
Indian Knowledge System (IKS)						
SIUCSIK111	Indian Knowledge System(IKS)	Indian Knowledge System	2	2		2
Total						22

Semester – II						
Course Code	Course Type	Course Title	Credits	Lectures/Week		
				Theory	Practical (2 lectures)	Total
SIUCSMJ121	Major Subject	Data Structures and fundamentals of Algorithm	3	3		3
SIUCSMJP121	Major Practical	Practical of SIUCSMJ121	1		1	1
SIUCSMN121	Minor Subject	Fundamentals of Mathematics and Statistics-II	3	3		3
SIUCSMNP121	Minor Practical	Practical of SIUCSMN121	1		1	1
Vocational and Skill Enhancement						
SIUCSVS121	Vocational Skill Course (VSC)	Programming with C	1	1		1
SIUCSVS121	Vocational Skill Course practical	Practical of SIUCSVS23	1		1	1
SIUCSSE121	Skill Enhancement Course (SEC)	Digital Electronics	1	1		1
SIUCSSE121	Skill Enhancement Course (SEC)	Practical of SIUCSSE24	1		1	1
Generic/ Open Elective Courses (OE)						
SIUCSOE121	Open Elective	Basics of R programming	2	2		2
SIUCSOE121	Open Elective Practical	Practical of SIUCSOE25	2		2	2
Ability Enhancement Courses (AEC)						
SIUCSAE121	Ability Enhancement Courses	Professional communication development- II	2	2		2
Value Education Courses (VEC)						
SIUCSVE121	Value Education Courses	Understanding India	2	2		2
Co- Curricular (CC)						
		NCC/ NSS/ Sports / Cultural/ Field project/ Industrial visit	2	2		2
Total						22

Semester I – Theory

Course	Title	Lectures	Credits
SIUCSMJ111	Programming with Python- I	3 per week (60 min per lec)	3
<p>Objectives The objective of this paper is to introduce various concepts of programming to the students using Python.</p> <p>Course Outcomes:</p> <ul style="list-style-type: none"> • CO1: Students should be able to understand the concepts of programming before actually starting to write programs. • CO2: Students should be able to develop logic for Problem Solving. • CO3: Students should be made familiar with the basic constructs of programming such as data, operations, conditions, loops, functions etc. • CO4: Students should be able to apply the problem solving skills using syntactically simple language i.e. Python (version: 3.X or higher) 			
Unit I	<p>Introduction: The Python Programming Language, History, features, Installing Python, Running Python program, Debugging : Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages</p> <p>Variables and Expressions: Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations. Conditional Statements: if, if-else, nested if –else Looping: for, while, nested loops Control statements: Terminating loops, skipping specific conditions.</p> <p>Compound Data types: Strings, Lists, Tuples, Dictionaries</p>	15 L	
Unit II	<p>Functions And Modules: Defining a function, calling a function, Advantages of functions, types of functions, function parameters, Formal parameters, Actual parameters, global and local variables, Anonymous functions, List comprehension Importing module, Creating & exploring modules.</p> <p>Anonymous functions. List comprehensions. Gentle introduction to object-oriented programming; using the built-in dir() function, enumerate the methods of strings, tuples, lists, dictionaries. Using these methods for problem-solving with compound types.</p> <p>Python File Input-Output: Opening and closing files, various types of file modes, reading and writing to files, manipulating directories. Iterables, iterators and their problem solving applications.</p>	15 L	

<p>Unit III</p>	<p>Exception handling: What is an exception, various keywords to handle exceptions such try, catch, except, else, finally, raise.</p> <p>Regular Expressions: Concept of regular expression, various types of regular expressions, using match function.</p> <p>Database connectivity in Python: Installing MySQL connector, accessing connector module, using connect, cursor, execute & close functions, reading single & multiple results of query execution, executing different types of statements, executing transactions, understanding exceptions in database connectivity.</p>	<p>15 L</p>
------------------------	---	-------------

Text books:

1. Magnus Lie Hetland, *Beginning Python: From Novice to Professional*, Apress
2. Paul Gries, et al., *Practical Programming: An Introduction to Computer Science Using Python 3*, Pragmatic Bookshelf, 2/E 2014

Additional References:

1. Charles Dierbach, *Introduction to Computer Science using Python*, Wiley, 2013
2. Paul Gries , Jennifer Campbell, Jason Montojo, *Practical Programming: An Introduction to Computer Science Using Python 3*, Pragmatic Bookshelf, 2/E 2014
3. Adesh Pandey, *Programming Languages – Principles and Paradigms*, Narosa, 2008

Course	Title	Lectures	Credits
SIUCSMJP111	Practicals on Programming with Python	2 per week (60 min per lec)	1
1	Installing and setting up the Python IDLE interpreter. Executing simple statements like expression statements (numeric and Boolean types), assert, assignment, delete statements; the print function for output, the input function.		
2	Programs based on conditional constructs(if, if else, if elif else, nested if)		
3	Programs based on for statement and the range function, using break and continue statements		
4	Programs based on the while statement		
5	Programs related to string manipulation		
5	Programs related to lists and list comprehensions		
6	Programs related to dictionaries		
7	Programs related to functions		
8	Programs to read and write files.		
9	Program to demonstrate exception handling.		
10	Program to demonstrate the use of regular expressions.		
11	Programs related to database handling		

Course	Title	Lectures	Credits
SIUCSMN111	Fundamentals of Mathematics and Statistics- I	3 per week (60 min per lec)	3
<p>Objectives: The purpose of the course is to familiarize the prospective learners with mathematical structures that are fundamentally discrete. This course introduces sets and functions, forming and solving recurrence relations and different counting principles. These concepts are useful to study or describe objects or problems in computer algorithms and programming languages.</p> <p>Course Outcomes:</p> <ul style="list-style-type: none"> • CO1: To provide an overview of the theory of discrete objects, starting with relations and partially ordered sets. • CO2: Study about recurrence relations, generating function and operations on them. • CO3: Give an understanding of graphs and trees, which are widely used in software. • CO4: Provide basic knowledge about models of automata theory and the corresponding formal languages. • CO5: Enable learners to know descriptive statistical concepts • CO6: Enable study of probability concept required for Computer learners 			
Unit I	<p>Recurrence Relations</p> <p>(a) Functions: Definition of function. Domain, co domain and the range of a function. Direct and inverse images. Injective, surjective and bijective functions. Composite and inverse functions.</p> <p>(b) Relations: Definition and examples. Properties of relations , Partial Ordering sets, Linear Ordering Hasse Diagrams , Maximum and Minimum elements, Lattices</p> <p>(c) Recurrence Relations: Definition of recurrence relations, Formulating recurrence relations, solving recurrence relations-Backtracking method, Linear homogeneous recurrence relations with constant coefficients. Solving linear homogeneous recurrence relations with constant coefficients of degree two when characteristic equation has distinct roots and only one root, Particular solutions of non linear homogeneous recurrence relation, Solution of recurrence relation by the method of generation functions, Applications-Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms.</p>	15L	
	<p>Counting Principles , Languages and Finite State Machine</p> <p>(a) Permutations and Combinations and counting Principles: Partition and Distribution of objects, Permutation with distinct and indistinct objects, Binomial numbers, Combination with identities: Pascal Identity, Vandermonde's Identity, Pascal triangle, Binomial theorem, Combination with indistinct objects.</p>		

	<p>(b) Counting Principles: Sum and Product Rules, Two-way counting, Tree diagram for solving counting problems, Pigeonhole Principle (without proof); Simple examples, Inclusion Exclusion Principle (Sieve formula) (Without proof).</p> <p>(c) Languages, Grammars and Machines: Languages , regular Expression and Regular languages, Finite state Automata, grammars, Finite state machines, Gödel numbers, Turing machines.</p>	
Unit II	<p>Graphs and Trees</p> <p>(a) Graphs : Definition and elementary results, Adjacency matrix, path matrix, Representing relations using digraphs, Warshall's algorithm-shortest path , Linked representation of a graph, Operations on graph with algorithms - searching in a graph; Insertion in a graph, Deleting from a graph, Traversing a graph- Breadth-First search and Depth-First search.</p> <p>(b) Trees: Definition and elementary results. Ordered rooted tree, Binary trees, Complete and extended binary trees, representing binary trees in memory, traversing binary trees, binary search tree, Algorithms for searching and inserting in binary search trees, Algorithms for deleting in a binary search tree.</p>	15L
Unit III	<p>(a) Data Presentation and Data Aggregation</p> <ul style="list-style-type: none"> ● Data types : attribute, variable, discrete and continuous variable ● Data presentation : frequency distribution, histogram to give, curves, stem and leaf display ● Measures of Central tendency: Mean, Median, mode for raw data, discrete, grouped frequency distribution. ● Measures dispersion: Variance, standard deviation, coefficient of variation for raw data, discrete and grouped frequency distribution, quartiles, quantiles Real life examples <p>(b) Moments, Skewness and Kurtosis</p> <ul style="list-style-type: none"> ● Moments: raw moments, central moments, relation between raw and central moments ● Measures of Skewness and Kurtosis: based on moments, quartiles, relation between mean, median, mode for symmetric, asymmetric frequency curve. <p>(c) Correlation and Regression: bivariate data, scatter plot, correlation, nonsense correlation, Karl pearson's coefficients of correlation, independence.</p> <ul style="list-style-type: none"> ● Linear regression: fitting of linear regression using least square regression, coefficient of determination, properties of regression coefficients (only statement) 	15L

Text books:

1. Trivedi, K.S.(2001) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

1. Ross, S.M. (2006): A First course in probability. 6th Edⁿ Pearson
2. Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): common statistical tests. Satyajeet Prakashan, Pune
3. Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
4. Gupta, S.C. and Kapoor, V.K. (1999): Applied Statistics, S. Chand and Son's, New Delhi
5. Montgomery, D.C. (2001): Planning and Analysis of Experiments, wiley.

Course	Title	Lectures	Credits
SIUCSMNP111	Practicals of Fundamentals of Mathematics and Statistics- I	2 per week (60 min per lec)	1
1	Write a Program on to implement a.Hasse Diagram b.Poset c.Lattice d.Tower of Hanoi		
2	Write a Program on to implement A. Permutation B.Combination C.Counting Principle		
3	Write a Program on to implement A. Breadth First Search B. Depth First Search		
4	Write a program on Binary Trees to perform A. Insertion B. Traversal		
5	Write a program to implement A. Basics functions B. Frequency Distribution and Data Presentation		
6	Write a program to implement Measure of Central Tendency A. Mean B. Median C. Mode		
7	Measure of Dispersion A. Standard Deviation B. Coefficient of Variance C.Quartiles		
8	Moments, Skewness and Kurtosis A. Central and Raw Moments B.Skewness C.Kurtosis		
9	Correlation and Regression A. Karl Pearson's coefficient of correlation B. Linear Regression		

Course	Title	Lectures	Credits
SIUCSVS111	Basic Web Programming	1 per week (60 min per lec)	1
<p>Objectives: The course has been designed to provide the basic knowledge for developing of the web pages using HTML,CSS and JavaScript programming language.</p> <p>Expected Learning Outcomes:</p> <ul style="list-style-type: none"> • CO1: Learn the fundamental technology used to define the structure of a webpage. • CO2: Understand the various platforms, devices, display resolutions, viewports, and browsers that render websites • CO3: To develop and implement client-side and server-side scripting language programs 			
Unit I	<p>HTML5: Fundamental Elements of HTML, Formatting Text in HTML, Organizing Text in HTML, Links and URLs in HTML, Tables in HTML, Images on a Web Page, Image Formats, Image Maps, Colors, FORMs in HTML, Interactive Elements, Working with Multimedia - Audio and Video File Formats, HTML elements for inserting Audio / Video on a web page</p> <p>CSS: Understanding the Syntax of CSS, CSS Selectors, Inserting CSS in an HTML Document, CSS properties to work with background of a Page, CSS properties to work with Fonts and Text Styles, CSS properties for positioning an element</p> <p>JavaScript: Using JavaScript in an HTML Document, Programming Fundamentals of JavaScript – Variables, Operators, Control Flow Statements, Popup Boxes, Functions – Defining and Invoking a Function, Defining Function arguments, Defining a Return Statement, Calling Functions with Timer, JavaScript Objects - String, RegExp, Math, Date, Browser Objects - Window, Navigator, History, Location, Document, Cookies, Document Object Model, Form Validation using JavaScript</p>	15L	
<p>Text Book(s):</p> <ol style="list-style-type: none"> 1) HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2ed, Dreamtech Press 2) Web Programming and Interactive Technologies, scriptDemics, StarEdu Solutions India. 3) PHP: A Beginners Guide, Vikram Vaswani, TMH <p>Additional Reference(s):</p> <ol style="list-style-type: none"> 1) HTML, XHTML, and CSS Bible Fifth Edition, Steven M. Schafer, WILEY 2) Learn to Master HTML 5, scriptDemics, StarEdu Solutions Pvt Ltd. 			

Course	Title	Lectures	Credits
SIUCSVS111	Practicals on Basic Web Programming	2 per week (60 min per lec)	1
1	A. Design a web page which displays data in a table B. Design a registration form web page		
2	Design a web page which contains three hyperlinks (audio,video, and gif image). <ul style="list-style-type: none"> • When a user clicks on an audio link, the web-page should open in the same tab with some audio content. • When a user clicks on a video web page should open in the same tab with some video content. • When a user clicks on a gif image web-page should open in the same tab with some gif content. • Every hyperlink web page should contain hyperlink (home). So that when user click on home it go back to home page(main page) 		
3	Design a webpage that makes use of Cascading Style Sheets with (Background, fonts, Text styles).		
4	A. Create a web page which takes a number from the user through the input box. onclick of button it should display the factorial of that number. B. Create a web page which takes series length from the user through the input box. Onclick of button it should display Fibonacci series of that length. C. Create a web page which takes a number from the user through input box. Onclick of button it should display the reverse of that number.		
5	Write a javascript program which contain following buttons i) browser window size (height and width), ii) current page details(hostname, protocol and port of the page), iii) browser details like(appversion, appname, language) . iv) Back v) Forward On click of first three buttons it should display the information.And onclick of back and forward button it should load the previous and next URL respectively.		
6	Write a javascript program to take firstname, last name ,age , contact-no, address for registration. Validate all the fields using Regular expression (RegExp object) such that i) firstname should contain only characters ii) lastname should contain only characters iii) age should contain only number iv) contact-no should contain only number v) address should not contain \$ sign		

Course	Title	Lectures	Credits
SIUCSSE111	Beginning MySQL	1 per week (60 min per lec)	1

Objectives:

The objective of this course is to introduce the concept of the DBMS with respect to the relational model, to specify the functional and data requirements for a typical database application and to understand creation, manipulation and querying of data in databases .

Expected Learning Outcomes

- **CO1:** Gain familiarity with the MySQL development environment
- **CO2:** Understand basic concepts of database development:
- **CO3:** SQL, Database design, Administration, and Security
- **CO4:** Design and code a database solution

Unit I	<p>Introduction to DBMS – Database, DBMS –Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture</p> <p>DDL Statements - Creating Databases, Using Databases, data types Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables, Backing Up and Restoring databases</p> <p>DML Statements – Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate function (count, min, max, avg, sum), group by clause, having clause</p> <p>Functions – String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse)</p> <p>Joining Tables – inner join, outer join (left outer, right outer, full outer)</p> <p>Subqueries – subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries</p> <p>Views (creating, altering dropping, renaming and manipulating views)</p> <p>DCL Statements (creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges)</p>	15L
---------------	--	-----

Text books:

1. Ramez Elmasri & Shamkant B.Navathe, Fundamentals of Database Systems, Pearson Education, Sixth Edition, 2010
2. Ramakrishnam, Gehrke, Database Management Systems, McGraw-Hill, 2007
3. Joel Murach,Murach's MySQL, Murach, 2012

Additional References:

1. Robert Sheldon, Geoff Moes, Begning MySQL, Wrox Press, 2005.

Course	Title	Lectures	Credits
SIUCSSE111	Practicals on Beginning MySQL	2 per week (60 min per lec)	1
1	<p>For given scenario</p> <p>Perform the following:</p> <ul style="list-style-type: none"> • Viewing all databases • Creating a Database • Viewing all Tables in a Database • Creating Tables (With and Without Constraints) • Inserting/Updating/Deleting Records in a Table • Saving (Commit) and Undoing (rollback) 		
2	<p>For given scenario</p> <p>Perform the following:</p> <ul style="list-style-type: none"> • Altering a Table • Dropping/Truncating/Renaming Tables • Backing up / Restoring a Database <p>Perform the following:</p> <ul style="list-style-type: none"> • Simple Queries • Simple Queries with Aggregate functions • Queries with Aggregate functions (group by and having clause) 		
3	<p>For given scenario</p> <p>Queries involving</p> <ul style="list-style-type: none"> • Date Functions • String Functions • Math Functions 		
4	<p>For given scenario</p> <p>Join Queries</p> <ul style="list-style-type: none"> • Inner Join • Outer Join <p>Subqueries</p> <ul style="list-style-type: none"> • With IN clause • With EXISTS clause 		
5	<p>For given scenario</p> <p>Views</p> <ul style="list-style-type: none"> • Creating Views (with and without check option) • Dropping views • Selecting from a view <p>DCL statements</p> <ul style="list-style-type: none"> • Granting and revoking permissions 		

Course:	Title	Lectures	Credits
SIUCSOE111	Basic Web Designing	2 per week (60 min per lec)	2
<p>Objectives: To provide insight into emerging technologies to design and develop state of - the art web applications using client-side scripting, server-side scripting, and database connectivity.</p> <p>Expected Learning Outcomes:</p> <ul style="list-style-type: none"> • CO1: To design valid, well-formed, scalable, and meaningful pages using emerging technologies. • CO2: Understand the various platforms, devices, display resolutions, viewports, and browsers that render websites • CO3: To develop and implement client-side and server-side scripting language programs. 			
Unit I	HTML5: Fundamental Elements of HTML, Formatting Text in HTML, Organizing Text in HTML, Links and URLs in HTML, Tables in HTML, Images on a Web Page, Image Formats, Image Maps, Colors, FORMs in HTML, Interactive Elements, Working with Multimedia - Audio and Video File Formats, HTML elements for inserting Audio / Video on a web page	15L	
Unit II	CSS: Understanding the Syntax of CSS, CSS Selectors, Inserting CSS in an HTML Document, CSS properties to work with background of a Page, CSS properties to work with Fonts and Text Styles, CSS properties for positioning an element Bootstrap: What is Bootstrap, containers-fixed container, fixed- width container, grid system, typography- display headings, Colors- text colors, background colours, tables,images, jumbotron, alerts, buttons.	15L	
<p>Text book:</p> <ol style="list-style-type: none"> 1. HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2ed, Dreamtech Press 2. Web Programming and Interactive Technologies, scriptDemics, StarEdu Solutions India. 3) PHP: A Beginners Guide, Vikram Vaswani, TMH <p>Additional References:</p> <ol style="list-style-type: none"> 1. HTML, XHTML, and CSS Bible Fifth Edition, Steven M. Schafer, WILEY 2. Learn to Master HTML 5, scriptDemics, StarEdu Solutions Pvt Ltd. 			

Course	Title	Lectures	Credits
SIUCSOE111	Practicals of Web Designing	4 per week (45 min per lec)	2
1	Design a web page which displays the map of India. Create a clickable region using an image map on the same image so that when we click on Maharashtra it opens another tab with information about Maharashtra. Create two more clickable regions for states of your choice.		
2	<p>A. Design a web page which contains three hyperlinks (audio, video, and gif image).</p> <p>I. When a user clicks on an audio link web page should open in the same tab with some audio content.</p> <p>II. When a user clicks on a video web page should open in the same tab with some video content.</p> <p>III. When a user clicks on a gif image web-page should open in the same tab with some gif content.</p>		
3	Design a webpage to display nested ordered and unordered lists.		
4	Design a webpage to display the time table of your class.		
5	Design a webpage to display student registration forms.		
6	Design a webpage that makes use of Cascading Style Sheets with (Background, fonts, Text styles).		
7	Create webpage to showcase bootstrap containers		
8	Create webpage to showcase bootstrap table with buttons and images		

Semester II - Theory

Course	Title	Lectures	Credits
SIUCSMJ121	Data Structures and fundamentals of Algorithm	3 per week (60 min per lec)	3
<p>Objectives: To explore and understand the concepts of Data Structures and its significance in programming. Provide a holistic approach to design, use and implement abstract data types. Understand the commonly used data structures and various forms of its implementation for different applications using Python.</p> <p>Expected Learning Outcomes:</p> <ul style="list-style-type: none"> • CO1: Learn about Data structures, its types and significance in computing • CO2: Explore about Abstract Data types and its implementation • CO3: Ability to program various applications using different data structure in Python 			
Unit I	<p>Abstract Data Types: Introduction, Bags, Iterators. Application</p> <p>Arrays: Array Structure, Python List, Two Dimensional Arrays, Matrix Abstract Data Type, Application</p> <p>Sets and Maps: Sets-Set ADT, Selecting Data Structure, List based implementation</p> <p>Maps: Map ADT, List Based Implementation, Application</p> <p>Searching and Sorting: Searching-Linear Search, Binary Search, Sorting-Bubble, Selection and Insertion Sort, Merge Sort, Quick Sort, Radix Sort</p>		15L
Unit II	<p>Linked Structures: Introduction, Singly Linked List-Traversing, Searching and Removing Nodes</p> <p>Stacks: Stack ADT, Implementing Stacks-Using Python List, Using Linked List, Stack Applications-Infix to postfix conversion, Evaluating Postfix Expressions</p> <p>Queues: Queue ADT, Implementing Queue-Using Python List, Circular Array, Using List, Priority Queues- Priority Queue ADT, Bounded and unbounded Priority Queues</p> <p>Advanced Linked List: Doubly Linked Lists-Organization and Operation, Circular Linked List-Organization and Operation</p>		15L

<p style="text-align: center;">Unit III</p>	<p>Recursion: Recursive Functions, Properties of Recursion, Its working, Recursive Applications</p> <p>Hash Table: Introduction, Hashing-Linear Probing, Clustering, Rehashing, Separate Chaining, Hash Functions</p> <p>Binary Trees: Tree Structure, Binary Tree-Properties, Implementation and Traversals, Heaps and Heapsort, Search Trees</p> <p>Graphs: Definition, Implementation in Python, Traversal algorithms (DFS and BFS)</p> <p>Algorithm Analysis: Introduction to algorithm, Why to analysis algorithm, Running time analysis, How to Compare Algorithms, Rate of Growth, Commonly Used Rates of Growth, Types of Analysis, Asymptotic Notation, Big-O Notation, Omega-Ω Notation, Theta-Θ Notation, Asymptotic Analysis, Properties of Notations, Commonly used Logarithms and Summations, Performance characteristics of algorithms, Master Theorem for Divide and Conquer, Master Theorem for Subtract and Conquer, Evaluating Python Code</p>	<p style="text-align: center;">15L</p>
<p>Text book:</p> <ol style="list-style-type: none"> 1) <i>Data Structure and algorithm Using Python</i>, Rance D. Necaie, 2016 Wiley India Edition 2) <i>Data Structure and Algorithm in Python</i>, Michael T. Goodrich, Robertom Tamassia, M. H. Goldwasser, 2016 Wiley India Edition <p>Additional References:</p> <ol style="list-style-type: none"> 1) <i>Data Structure and Algorithmic Thinking with Python</i>- Narasimha Karumanchi, 2015, Careermonk Publications 2) <i>Fundamentals of Python: Data Structures</i>, Kenneth Lambert, Delmar Cengage Learning 		

Course	Title	Lectures	Credits
SIUCSMJP121	Practicals of Data Structures and fundamentals of Algorithm	2 per week (60 min per lec)	1
1	Implement Linear Search to find an item in a list.		
2	Implement binary search to find an item in an ordered list.		
3	Implement Sorting Algorithms A. Bubble sort B. Insertion sort		
4	Implement use of Sets and various operations on Sets.		
5	Implement working of Stacks. (pop method to take the last item added off the stack and a push method to add an item to the stack)		
6	Implement Program for A. Infix to Postfix conversion B. Postfix Evaluation		
7	Implement the following A. A queue as a list which you add and delete items from. B. A circular queue. (The beginning items of the queue can be reused).		
8	Implement Linked list and demonstrate the functionality to add and delete items in the linked list.		
9	Implement Binary Tree and its traversals.		
10	Recursive implementation of A. Factorial B. Fibonacci		
11	Write a Python program to sort n names using Quick sort algorithm. Discuss the complexity of algorithms used.		
12	Write a Python program to sort n numbers using Merge sort algorithm. Discuss the complexity of algorithms used.		
13	Write a Python program for inserting an element into a binary tree.		
14	Write a Python program for deleting an element (assuming data is given) from a binary tree.		

Course	Title	Lectures	Credits
SIUCSMN121	Fundamentals of Mathematics and Statistics-II	3 per week (60 min per lec)	3
<p>Objective: The objective of this paper is to explore the style of structured programming to give the idea to the students how programming can be used for designing real-life applications by reading/writing to files, GUI programming, interfacing database/networks and various other features.</p> <p>Expected Learning Outcomes</p> <ul style="list-style-type: none"> • CO1: Students should be able to understand how to read/write to files using python. • CO2: Students should be able to catch their own errors that happen during execution of programs. • CO3: Students should get an introduction to the concept of pattern matching. • CO4: Students should be made familiar with the concepts of GUI controls and designing GUI applications. • CO5: Students should be able to connect to the database to move the data to/from the application. • CO6: Students should know how to connect to computers, read from URL and send email. 			
Unit I	<p>Derivatives And Its Applications: Review of Functions, limit of a function, continuity of a function, derivative function. Derivative In Graphing And Applications: Analysis of Functions: Increase, Decrease, Concavity, Relative Extrema; Graphing Polynomials, Rational Functions, Cusps and Vertical Tangents. Absolute Maxima and Minima, Applied Maximum and Minimum Problems, Newton's Method.</p>	15 L	
Unit II	<p>Integration And Its Applications: An Overview of the Area Problem, Indefinite Integral, Definition of Area as a Limit; Sigma Notation, Definite Integral, Evaluating Definite Integrals by Substitution, Area Between Two Curves, Length of a Plane Curve. Numerical Integration: Simpson's Rule. Modeling with Differential Equations, Separation of Variables, Slope Fields, Euler's Method, FirstOrder Differential Equations and Applications.</p> <p>Partial Derivatives And Its Applications: Functions of Two or More Variables Limits and Continuity Partial Derivatives, Differentiability, Differentials, and Local Linearity, Chain Rule, Directional Derivatives and Gradients, Tangent Planes and Normal, Vectors, Maxima and Minima of Functions of Two Variables.</p>	15 L	
Unit III	<p>Standard distributions: random variable; discrete, continuous, expectation and variance of a random variable, pmf, pdf, cdf, reliability, Introduction and properties without proof for following distributions; binomial, normal, chi-square, t, F. Examples</p>	15 L	

	<p>Hypothesis testing: one sided, two sided hypothesis, critical region, p-value, tests based on t, Normal and F, confidence intervals. Analysis of variance : one-way, two-way analysis of variance</p> <p>Non-parametric tests: need of non-parametric tests, sign test, Wilcoxon's signed rank test, run test, Kruskal-Wallis tests. Post-hoc analysis of one-way analysis of variance : Duncan's test Chi-square test of association</p>	
--	--	--

Text Book:

1. Trivedi, K.S.(2009) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

1. Ross, S.M. (2006): A First course in probability. 6th Edⁿ Pearson
2. Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): Common statistical tests. Satyajeet Prakashan, Pune
3. Gupta, S.C. and Kapoor, V.K. (2002) : Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
4. Gupta, S.C. and Kapoor, V.K. (4th Edition) : Applied Statistics, S. Chand and Son's, New Delhi
5. Montgomery, D.C. (2001): Planning and Analysis of Experiments, Wiley.

Course	Title	Lectures	Credits
SIUCSMNP121	Practicals of Fundamentals of Mathematics and Statistics- II	2 per week (60 min per lec)	1
1	Increasing, decreasing, concave up and concave down functions		
2	Relative maxima, relative minima, absolute maxima, absolute minima		
3	Newton's method to find approximate solution of an equation		
4	Numerical integration using Simpson's rule		
5	Solution of a differential equation, Euler's method, Runge Kutta Method.		
6	Calculation of Partial derivatives of functions		
7	Maxima and minima of functions of two variables		
8	Problems based on binomial distribution		
9	Problems based on normal distribution		
10	Parametric test		
11	Non parametric tests- I		
12	Non- Parametric tests – II		

Course	Title	Lectures	Credits
SIUCSVS121	Programming with C	1 per week (60 min per lec)	1
<p>Objectives: The objective of this course is to provide a comprehensive study of the C programming language, stressing strengths of C, which provide the students with the means of writing modular, efficient, and portable code.</p> <p>Course Outcome:</p> <ul style="list-style-type: none"> ● CO1: Students should be able to write, compile and debug programs in C language. ● CO2: Students should be able to use different data types in a computer program. ● CO3: Students should be able to design programs involving decision structures, loops and functions. ● CO4: Students should be able to explain the difference between call by value and call by reference. ● CO5: Students should be able to understand the dynamics of memory by the use of pointers. 			
Unit I	<p>Structure of C program: Header and body, Use of comments. Interpreters vs compilers, Python vs C. Compilation of a program. Formatted I/O: printf(), scanf().</p> <p>Data: Variables, Constants, data types like: int, float char, double and void, short and long size qualifiers, signed and unsigned qualifiers.</p> <p>Variables: Declaring variables, scope of the variables according to block, hierarchy of data types.</p> <p>Iterations: Control statements for decision making: (i) Branching: if statement, else.. if statement, switch statement. (ii) Looping: while loop, do.. while, for loop. (iii) Jump statements: break, continue and goto.</p> <p>Arrays: (One and two dimensional), declaring array variables, initialization of arrays, accessing array elements.</p> <p>Functions: Function declaration, function definition, Global and local variables, return statement, Calling a function by passing values.</p> <p>Recursion: Definition, Recursive functions.</p> <p>Pointer: Fundamentals, Pointer variables, Referencing and dereferencing, Pointer Arithmetic, Using Pointers with Arrays, Using Pointers with Strings, Array of Pointers, Pointers as function arguments, Functions returning pointers.</p>	15L	

	<p>Dynamic Memory Allocation: malloc(), calloc(), realloc(), free() and sizeof operator.</p> <p>Structure: Declaration of structure, reading and assignment of structure variables, Array of structures, arrays within structures, structures within structures. Compare C structures with Python tuples.</p>	
<p>Text books:</p> <ol style="list-style-type: none">1. Programming in ANSI C (Third Edition) : E Balagurusamy, TMH <p>Additional References:</p> <ol style="list-style-type: none">1. Pradip Dey, Manas Ghosh, "Programming in C", second edition, Oxford University Press2. Yashavant P. Kanetkar. " Let Us C", BPB Publications		

Course	Title	Lectures	Credits
SIUCSVS121	Practicals Of Programming with C	2 per week (60 min per week)	1
1	Basic Programs(Variables, Operators): <ol style="list-style-type: none"> Write a program to find the addition, subtraction, multiplication and division of two numbers. Write a program to find the area of rectangle, square and circle. Write a program to find the volume of a cube, sphere, and cylinder. 		
2	<ol style="list-style-type: none"> Programs to demonstrate data input and output functions Programs to manipulate strings 		
3	Conditional statements and loops <ol style="list-style-type: none"> Write a program to check whether the number is even or odd. Write a program to check whether the number is positive, negative or zero. Write a program to find the sum of squares of digits of a number. Write a program to reverse the digits of an integer. 		
4	Programs on Functions.		
5	Recursive functions <ol style="list-style-type: none"> Write a program to find the factorial of a number using a recursive function. Write a program to find the sum of natural numbers using a recursive function. 		
6	Arrays <ol style="list-style-type: none"> Write a program to find the largest value that is stored in the array. Write a program using pointers to compute the sum of all elements stored in an array. Write a program to arrange the 'n' numbers stored in the array in ascending and descending order. 		
7	Pointers <ol style="list-style-type: none"> Write a program to demonstrate the use of pointers. Write a program to perform addition and subtraction of two pointer variables. 		
8	Programs on structures.		
9	Programs on unions.		
10	Programs on File Handling <ol style="list-style-type: none"> Write a program to Create a File, Write in it, And Close the File. Write a program to Open a File, Read from it, And Close the File Write a program to read the name and marks of 'n' number of students and store them in a file. 		

Course	Title	Lectures	Credits
SIUCSSE121	Digital Electronics	1 per week (60 min per lec)	1
<p>Objectives: The objective of this course is to provide a comprehensive study of the C programming language, stressing strengths of C, which provide the students with the means of writing modular, efficient, and portable code.</p> <p>Course Outcomes:</p> <ul style="list-style-type: none"> • CO1: Apply concepts of Digital Binary System and implementation of Gates. • CO2: Analyze and design of Combinational logic circuits. • CO3: Analyze and design of Sequential logic circuits with their applications. • CO4: Implement the Design procedure of Synchronous & Asynchronous Sequential Circuits. • CO5: Apply the concept of Digital Logic Families with circuit implementation. 			
Unit I	<p>Computer Abstractions and Technology: Basic structure and operation of a computer, functional units and their interaction.</p> <p>Number Systems: Binary, Decimal, Octal and Hexadecimal.</p> <p>Logic circuits and functions: Combinational circuits and functions: Basic logic gates and functions, truth tables; logic circuits and functions. Minimization with Karnaugh maps. Synthesis of logic functions with and-or-not gates, nand gates, nor gates. Fan-in and fan-out requirements; tristate buffers. Half adder, full adder, ripple carry adder. (Flip flops) Gated S-R and D latches, edge-triggered D latch. Shift registers and registers. Decoders, multiplexers. Sequential circuits and functions: State diagram and state table</p> <p>Instruction set architectures: Memory organization, addressing and operations; word size, big-endian and little endian arrangements. Instructions, sequencing. Instruction sets for RISC and CISC (examples Altera NIOS II and Freescale ColdFire). Operand addressing modes; pointers; indexing for arrays. Machine language, assembly language, assembler directives. Function calls, processor runtime stack, stack frame.</p> <p>Types of machine instructions: arithmetic, logic, shift, etc. Instruction sets, RISC and CISC examples.</p> <p>Basic Processor Unit: Main components of a processor: registers and register files, ALU, control unit, instruction fetch unit, interfaces to instruction and data memories. Datapath. Instruction fetch and execute; executing arithmetic/logic, memory access and branch instructions; hardwired and microprogrammed control for RISC and CISC.</p>		15L

	Basic I/O: Accessing I/O devices, data transfers between processor and I/O devices. Interrupts and exceptions: interrupt requests and processing.	
--	--	--

Text book:

1. Carl Hamacher et al., Computer Organization and Embedded Systems, 6 ed., McGraw-Hill 2012

Additional References:

1. Patterson and Hennessy, Computer Organization and Design, Morgan Kaufmann, ARM Edition, 2011
2. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd. , 4th Edition, 2010

Course	Title	Lectures	Credits
SIUCSSE121	Practical on Digital Electronics	2 per week (60 min per lec)	1
1	Study and verify the truth table of various logic gates (NOT, AND, OR, NAND, NOR, EX-OR, and EX-NOR).		
2	Simplify given Boolean expression and realize it		
3	Design and verify a half/full adder		
4	Design and verify half/full subtractor		
5	Verify the operation of a UP and DOWN Counter.		
6	Verify the operation of a 4 bit shift register		
7	Design and verify the operation of flip-flops (SR, D and T) using logic gates.		
8	Using SPIM, write and test an adding machine program that repeatedly reads in integers and adds them into a running sum. The program should stop when it gets an input that is 0, printing out the sum at that point		
9	Using SPIM, write and test a program that reads in a number from the user and prints if it is prime or not prime		

Course	Title	Lectures	Credits
SIUCSOE121	Basics of R programming	2 per week (60 min per lec)	2

Objective:

The course covers data reading and its manipulation using R, which is widely used for data analysis internationally. The course also covers different control structures and design of user-defined functions. Loading, installing and building packages are covered.

Course Outcome:

- CO1: Develop an R script and execute it.
- CO2: Install, load and deploy the required packages, and build new packages for sharing and reusability.
- CO3: Extract data from different sources using API and use it for data analysis.
- CO4: Visualize and summarize the data.
- CO5: Design application with database connectivity for data analysis.

Unit I	<p>Introduction: R interpreter, Introduction to major R data structures like vectors, matrices, arrays, list and data frames, Control Structures, vectorized if and multiple selection, functions.</p> <p>Installing, loading and using packages: Read/write data from/in files, extracting data from web-sites, Clean data, Transform data by sorting, adding/removing new/existing columns, centring, scaling and normalizing the data values, converting types of values, using string in-built functions.</p>	15L
Unit II	<p>Statistical analysis of data - for summarizing and understanding data, Visualizing data using scatter plot, line plot, bar chart, histogram and box plot.</p>	15L

Textbook:

1. Cotton, R., Learning R: a step by step function guide to data analysis. 1st edition. O'reilly Media Inc

Additional References:

1. Gardener, M.(2017). Beginning R: The statistical programming language, WILEY Lawrence, M., & Verzani, J. (2016).
2. Programming Graphical User Interfaces in R. CRC press. (ebook)

Course	Title	Lectures	Credits																								
SIUCSOE121	Practicals of Basics of R Programming	4 per week (45 min per lec)	2																								
1	Write a program to check whether a year (integer) entered by the user is a leap year or not?																										
2	Write an R program to find the sum of natural numbers without formula using the if-else statement and the while loop																										
3	Write a program that prints the grades of the students according to the marks obtained. The grading of the marks should be as follows.																										
	<table border="1"> <thead> <tr> <th>Marks</th> <th>Grades</th> </tr> </thead> <tbody> <tr> <td>800-1000</td> <td>A+</td> </tr> <tr> <td>700 – 800</td> <td>A</td> </tr> <tr> <td>500 – 700</td> <td>B+</td> </tr> <tr> <td>400-500</td> <td>B</td> </tr> <tr> <td>150 – 400</td> <td>C</td> </tr> <tr> <td>Less than 150</td> <td>D</td> </tr> </tbody> </table>			Marks	Grades	800-1000	A+	700 – 800	A	500 – 700	B+	400-500	B	150 – 400	C	Less than 150	D										
Marks	Grades																										
800-1000	A+																										
700 – 800	A																										
500 – 700	B+																										
400-500	B																										
150 – 400	C																										
Less than 150	D																										
4	Write a set of instructions to create the following matrix using vectors and rbind() function. Rename the rows to Lang1,Lang2 & Lang3 respectively and use the function to access any one element using row names.																										
	<table border="1"> <thead> <tr> <th rowspan="2">Rows</th> <th colspan="4">Columns</th> </tr> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>C#</td> <td>Java</td> <td>Cobol</td> <td>.Net</td> </tr> <tr> <td>2</td> <td>JavaScript</td> <td>NodeJs</td> <td>R</td> <td>Azure</td> </tr> <tr> <td>3</td> <td>Power BI</td> <td>ASP.Net</td> <td>Unity</td> <td>Block Chain</td> </tr> </tbody> </table>			Rows	Columns				1	2	3	4	1	C#	Java	Cobol	.Net	2	JavaScript	NodeJs	R	Azure	3	Power BI	ASP.Net	Unity	Block Chain
Rows	Columns																										
	1	2	3	4																							
1	C#	Java	Cobol	.Net																							
2	JavaScript	NodeJs	R	Azure																							
3	Power BI	ASP.Net	Unity	Block Chain																							
5	Write an R script to do the following: a) simulate a sample of 100 random data points from a normal distribution with mean 100 and standard deviation 5 and store the result in a vector. b) visualize the vector created above using different plots.																										
6	In the library MASS is a dataset UScereal which contains information about popular breakfast cereals. Attach the data set and use different kinds of plots to investigate the following relationships: a) relationship between manufacturer and shelf b) relationship between fat and vitamins c) relationship between fat and shelf d) relationship between carbohydrates and sugars e) relationship between fibre and manufacturer f) relationship between sodium and sugars																										
7	Using the Algae data set from package DMwR to complete the following tasks. a) create a graph that you find adequate to show the distribution of the values of algae a6. b) show the distribution of the values of size 3.																										

	<ul style="list-style-type: none"> c) check visually if oPO4 follows a normal distribution. d) produce a graph that allows you to understand how the values of NO3 are distributed across the sizes of rivers. e) using a graph check if the distribution of algae a1 varies with the speed of the river. f) visualize the relationship between the frequencies of algae a1 and a6. Give the appropriate graph title, x-axis and y-axis title.
8	<p>Let us use the built-in dataset air quality which has Daily air quality measurements in New York, May to September 1973. Create a histogram by using appropriate arguments for the following statements.</p> <ul style="list-style-type: none"> a) Assigning names, using the air quality data set. b) Change colors of the Histogram c) Remove Axis and Add labels to Histogram d) Change Axis limits of a Histogram e) Create a Histogram with density and Add Density curve to the histogram

Evaluation Scheme

Semester – I							
	Credits		Marks		Distribution		
	Theory	Practical	Theory	Practical			
Subject 1	3	1	75	25	Sem end: 50	Internal :25	Practical: 25
Subject 2	3	1	75	25	Sem end: 50	Internal :25	Practical: 25
OE	2	2	50	50	Sem end: 50	Internal: 50	
VSC	1	1	20	30	Internal: 20	Practical:30	
SEC	1	1	20	30	Internal: 20	Practical:30	
IKS	2	-	50	-	Sem end: 30	Internal :20	
AEC	2	-	50	-	Sem end: 30	Internal :20	
VEC	2	-	50	-	Sem end: 30	Internal :20	
Total							22

Semester – II							
	Credits		Marks		Distribution		
	Theory	Practical	Theory	Practical			
Subject 1	3	1	75	25	Sem end: 50	Internal :25	Practical: 25
Subject 2	3	1	75	25	Sem end: 50	Internal :25	Practical: 25
OE	2	2	50	50	Sem end: 50	Internal: 50	
VSC	1	1	50		Internal: 20	Practical:30	
SEC	1	1	50		Internal: 20	Practical:30	
AEC	2	-	50	-	Sem end: 30	Internal :20	
VEC	2	-	50	-	Sem end: 30	Internal :20	
CC	2	-	50	-			
Total							22