PUNYASHLOK AHILYADEVI HOLKAR SOLAPUR UNIVERSITY, SOLAPUR

पुण्यश्लोक अहिल्यादेवी होळकर
सोलापूर विद्यापीठ

॥ विद्यया संपन्नता ॥

NAAC Accredited-2015
‘B’ Grade (CGPA 2.62)

Name of the Faculty: Science & Technology

CHOICE BASED CREDIT SYSTEM

Syllabus

Name of the Course: B.Sc. (Entire Computer Science) - III (Semester –V and VI)

(Syllabus to be implemented w.e.f. June 2021)

Preamble : B. Sc. (Entire Computer Science) is a 3 year undergraduate programme with a specialization in the domain of computer science, software and hardware related aspects. B.Sc. (ECS) programme is perfect for students who want to make a career in computers. Major subjects in this programme include Digital Electronics, Computer Programming Theory, Discrete Mathematics, Advanced Programming using Python, C++, Java, etc. The course curriculum is inclusive of theory and practical which makes the students well trained and skillful in the field of programming, software, and network.

Objective of the programme :

- To develop problem solving abilities using a computer.
- To build the necessary skill set and analytical abilities for developing computer based solutions for real life problems.
- To train students in professional skills related to Software Industry.
- To prepare necessary knowledge base for research and development in Computer Science.
- To help students build-up a successful career in Computer Science and to produce entrepreneurs who can innovate and develop software products.

Programme Outcome : B.Sc. (ECS) programme has been designed to prepare graduates for attaining the following specific outcomes:

- An ability to apply knowledge of mathematics, statistics and computer science in practice.
- An ability to enhance not only comprehensive understanding of the theory but its application too in diverse field.
- The program prepares the young professional for a range of computer applications, computer organization, techniques of Computer Networking, Software Engineering, Web development, Database management and Advance Java
- An ability to design a computing system to meet desired needs within realistic constraints such as safety, security and applicability in multidisciplinary teams with positive attitude.
- In order to enhance programming skills of the young IT professionals, the program has introduced the concept of project development in each language/technology learnt during curriculum.

Eligibility for B.Sc. (ECS) Part-I :

i) The candidate passing the Higher Secondary Examination Conducted by the Maharashtra State Board of Higher Secondary Education, with science stream MCVC with Science Subject D. Pharm, Diploma, Engineering, Agricultural Diploma, Diary Diploma shall be allowed to enter upon the B. Sc. Part-I Course. OR

ii) An examination of any other statutory University or an Examining Body recognized as equivalent thereto.

iii) Repeater Student will be allowed to take fresh admission to the same Class with same subjects or different subjects.

Faculty of Science and Technology
Syllabus of B.Sc. (ECS)-III (CBCS) (w.e.f. 2021-22)

Subject/ Core Course	Name and Type of the Paper		No. of Papers/ Practical	Hrs./ Week			Total Marks per Paper	UA	CA	Credits
	Type	Name		L	T	P				
Class:	**B.Sc. (Entire Computer Science)- III Semester – V**									
Ability Enhancement Course	Paper – II Part A	English (Business English)	Paper II Part A	4	--	--	50	40	10	2.0
Core	DSE 1 A	Data Communication and Networking	Paper IX	4	--	--	100	80	20	4.0
	DSE 2 A	Theory of Computer Science	Paper X	4	--	--	100	80	20	4.0
	DSE 3 A	Visual Programming	Paper XI	4	--	--	100	80	20	4.0
	DSE 4 A	Advanced Java	Paper XII	4	--	--	100	80	20	4.0
Skill Enhancement Course	SEC 3	Advanced Python Programming	Paper XIII	4	--	--	100	80	20	4.0
Total Theory Semester-V				**24**			**550**	**440**	**110**	**22**
Class:	**B.Sc. (Entire Computer Science)- III Semester – VI**									
Ability Enhancement Course	Paper – II Part B	English (Business English)	Paper II Part B	4	--	--	50	40	10	2.0
Core	DSE 1 B	System Security	Paper XIV	4	--	--	100	80	20	4.0
	DSE 2 B	Compiler Construction	Paper XV	4	--	--	100	80	20	4.0
	DSE 3 B	Internet Programming using ASP.Net	Paper XVI	4	--	--	100	80	20	4.0
	DSE 4 B	Angular JS	Paper XVII	4	--	--	100	80	20	4.0
Skill Enhancement Course	SEC 4	Mobile Application Development	Paper XVIII	4	--	--	100	80	20	4.0
Total Theory Semester-VI				**24**			**550**	**440**	**110**	**22**
Practical's on		DSE 2 A and DSE 2 B	Practical IV	--	--	5	100	80	20	4.0
		DSE 3 A and DSE 3 B	Practical V	--	---	5	100	80	20	4.0
		DSE 4 A and DSE 4 B	Practical VI	--	--	5	100	80	20	4.0
		Project work	Practical VII	--	--	5	100	80	20	4.0
Total (practical's)						**20**	**400**	**320**	**80**	**16**
Grand Total				**48**	**--**	**20**	**1500**	**1200**	**300**	**60**

Abbreviations:

L: Lectures, T: Tutorials, P: Practical, UA : University Assessment, CA : College Assessment, CC: Core Course, AEC : Ability Enhancement Course, DSE : Discipline Specific Elective Paper, SEC : Skill Enhancement Course

Skill Based Add – On Credit Course to be conducted in Sem. V

Name of the Paper	Hrs. / Week	Total Marks	UA	CA	Credit
SWAYAM or NPTEL or MOOC Course / Internship / Industrial Training / Courses offered by College	04	100[#]	--	100[#]	4.0

1. # - Not for SWAYAM or NPTEL or MOOC course.
2. In case of SWAYAM / NPTEL course credits should be given by SWAYAM / NPTEL / MOOC as per UGC rules.
3. If students complete SWAYAM / NPTEL / MOOC course, their marks/grades should be submitted by the college to the university.
4. Add-on College Course list should be submitted to the University for information

Type: DSE 1 A

Course Title: Data Communication and Networking

(Paper Code: Paper IX)

Course Objectives:

1. To understand the structure of Data Communications System and its components.
2. Be familiarize with different network terminologies.

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Familiarize with contemporary issues in network technologies.
2. Know the layered model approach explained in OSI and TCP/IP network models
3. Identify different types of network devices and their functions within a network.
4. Know the Basic routing mechanisms, IP addressing scheme and internetworking concepts.
5. Familiarize with IP and TCP Internet protocols.
6. Understand major concepts involved in design of WAN, LAN and wireless networks.
7. Know the basics of network configuration and maintenance.
8. Know the fundamentals of network security issues.

Unit 1: Introduction to Computer Networks [6]

Network Definition, Network Topologies, Network Classifications, Network Protocol, Layered Network Architecture, Overview of ISO-OSI Reference Model, Overview of TCP/IP Protocol Suite.

Unit 2: Data Communication Fundamentals and Techniques [10]

Signals-Analog and Digital Signal, Data-Rate Limits, Digital to Digital Line Encoding Schemes, Pulse Code Modulation, Parallel and Serial Transmission, Digital to Analog Modulation, Multiplexing Techniques- FDM, TDM, Transmission Media, Switching: Circuit Switching, Message Switching, Packet Switching,

Unit 3: Data Link Layer Functions and Protocols [8]

Design issues, Error Detection and Error Correction Techniques, Data-Link Control- Framing and Flow Control, Error Recovery Protocols-Stop and Wait ARQ, Go-Back-N ARQ, Point to Point Protocol on Internet.

Unit 4: Multiple Access Protocol and Network Layer [8]

Design issues, CSMA/CD Protocols, Ethernet LANS; Connecting LAN and Back -Bone Networks- Repeaters, Hubs, Switches, Bridges, Router and Gateways, Networks Layer Functions and Protocols, Routing, Routing Algorithms, Network Layer Protocol of Internet - IP Protocol, Internet Control Protocols.

Unit 5: Transport, Session, Presentation and Application Layer Protocol [12]

Transport Services- Error and Flow Control, Connection Establishment and Connection Release, Flow Control & Buffering, TCP/IP protocol suite, Concept of- TCP, UDP, IP, FTP, DNS, Telnet, SMTP, POP, HTTP, WWW, ARP, RARP.

Reference Books:

- B. A. Forouzan: Data Communications and Networking, Fourth edition, THM Publishing Company Ltd 2007.
- S. Tanenbaum: Computer Networks, Fourth edition, PHI Pvt. Ltd 2002

Type: DSE 2 A

Course Title: Theory of Computer Science

(Paper Code: Paper X)

Course Objectives

1. Course should provide a formal connection between algorithmic problem solving and the theory of languages and automata and develop them into a mathematical (and less magical) view towards algorithmic design and in general computation itself.
2. The course should in addition clarify the practical view towards the applications of these ideas in the engineering part of CS.

Course Outcomes: After learning this course, the students should be able to-

1. Understand the basic concepts and application in Theory of Computation.
2. Apply this basic knowledge of Theory of Computation in the computer field to solve computational problems and in the field of compiler also.

Unit 1: Preliminaries **[4]**

Basic Definitions, Sets, Various ways of describing a Set, Subsets, operations on Sets, Infinite Sets Relations, Properties of relations, Equivalence of relations.

Unit 2: Finite Automata **[10]**

Introduction, Deterministic Finite Automata, Non Deterministic Finite Automata, The Equivalence of DFAs and NFAs, Finite Automata with ε Moves, Equivalence of NFA with ε Transitions and NFA without Transitions, Finite Automata with output, Moore Machine, Melay Machine Equivalence of Moore and Melay Machine.

Unit 3: Regular Expression and Properties of Regular Sets **[10]**

Regular Expression Operations on set of strings, Regular Expression, Regular Sets, Equivalence of finite automata and regular expression Properties of Regular Sets Closure properties, The pumping lemma of regular sets, Application of pumping lemma.

Unit 4: Regular and Context Free Grammars **[8]**

Context Free Grammars (CFG) Derivation and Language generated by grammar, Derivation Trees, Ambiguity of CFG, Simplification of CFG, Normal forms of CFG Regular Grammars Equivalence of regular grammars and finite automata Closure properties of CFG.

Unit 5: Pushdown Automata [8]

Introduction, Definitions, Equivalence of acceptance by final state and empty stack, Definition of DPDA and NPDA their correlation and examples of NPDA, CFG to PDA: Method and example, Closure properties of Regular language, Application of PDA.

Unit 6: Introduction of Turing Machine [4]

Turing Machine model and definition of TM, Language accepted by TM, Design of TM and examples.

Reference Books:

- J.P. Hopcroft, Rajeev Motwani, J.D. Ullman, Introduction to Automata Theory, Languages and Computation, II Edition, Pearson Education, 2001.
- John Martin, Introduction to Languages and Theory of Computation, Tata McGrawHill, 2003.
- Daniel I.A., Cohen, Introduction to Computer Theory, 2 nd Edition, John Wiley and Sons, Inc, 2000.

Type: DSE 3 A

Course Title: Visual Programming

(Paper Code: Paper XI)

Course Objectives:

1. To understand .Net Framework.
2. To learn computer programming using the Visual programming language with object-oriented programming principles.
3. To understand the concepts of Exception handling mechanism and Input output programming paradigms.
4. Emphasis is on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger.

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Design, create, build, and debug Visual programming applications.
2. Explore Visual Programming Integrated Development Environment (IDE).
3. Implement syntax rules in Visual programming.
4. Explain variables and data types used in program development.
5. Write and apply decision structures for determining different operations.
6. Write and apply loop structures to perform repetitive tasks.
7. Write and apply procedures, sub-procedures, and functions to create manageable code.

Unit 1: Introduction to Dot.Net Framework **[6]**

Introduction to DOTNET, DOT NET class framework, Common Language Runtime, Overview, Elements of .NET application, Memory Management, Garbage Collector: Faster Memory allocation, Optimizations, Common Language Integration, Common type system, User and Program Interface

Unit 2: Introduction to C# **[6]**

C# Language elements, Data types -Reference Type and Value Type, Boxing and Unboxing, Enum and Constant, Operators, Control Statements, working with Arrays and Strings, Parameter passing technique: Pass by value and by reference, out parameters, Variable length parameter.

Unit 3: Object oriented Programming Concepts [8]

Working with Indexer and Properties, Constructor and Destructor, working with "static" Members, Inheritance & Polymorphism - Types of Inheritance - Constructor in Inheritance - Interface Implementation - Operator and method Overloading and overriding - Static and Dynamic Binding and Virtual Methods, Abstract Class, sealed keyword.

Unit 4: Exception Handling and I/O Programming [8]

What is Exception, Rules for Handling Exception, Exception classes, Exception handling keywords, Throwing exceptions, Stream Classes, System.IO and Base classes of Stream, Console I/O Stream, Working with File, Directory classes.

Unit 5: Delegates, Collection classes [6]

Introduction to Delegation, Types of delegates, Anonymous Methods, ArrayList, HashTable, Stack, Queue, Writing custom generic classes, working with Generic Collection Classes.

Unit 6: Windows Forms and ADO.NET [10]

Controls: Common control Group, Data, control Group, Dialog control Group, Container control Group, Menus and Context Menus: Menu Strip, Toolbar Strip. SDI and MDI Applications, Evolution of ADO.NET, Connected and Disconnect Classes, Establishing Connection with Database, Executing simple Insert, Update and Delete, Statements, DataReader and DataAdapter, Dataset and its Advantages, Stored Procedures.

Reference Books:

- “Programming C#”- Jesse Liberty, O’Reilly Press.
- “Professional C#”-Robinson et al, Wrox Press, 2002.
- “The Complete Reference: C#”-Herbert Schildt, Tata McGraw Hill.
- “The Complete Reference: Ado.Net”- Jerke, Tata McGraw Hill.
- “C# for programmer”-Deilte-Pearson

Type: DSE 4 A

Course Title: Advanced Java

(Paper Code: Paper XII)

Course Objectives:

1. To learn GUI programming using swing Technology
2. To study database programming using Java.
3. To study web development concept using Servlet and JSP
4. To learn socket programming concept using Networking.

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Design, create, build, and debug Java applications.
2. Explore Integrated Development Environment (IDE).

Unit 1: Networking [6]

Basics, networking classes and interfaces, using java.net package, doing TCP/IP and Datagram Programming.

Unit 2: Introduction to Swing Technology [10]

JApplet, JFrame and JComponent, Icons and Labels, Handling Threading issues, Text fields, JButton class, Check Boxes, Radio buttons, Combo boxes, Tabbed panes, Scroll panes, Tree, Table and Menus.

Unit 3: Working with databases [6]

Steps for Connecting to databases, Types of Drivers, Handling Exceptions, Creating and Using Statement Objects, Using Statements to Insert, Update, Delete Data into a Database, Using the ResultSet Class, Data navigation, Prepared Statements, Callable Statements.

Unit 4: Servlets [10]

Introduction, HTTP Request Model, Sending the HTTP Request, HTTP Request Methods, Servlet Architecture, Servlets Packages, Life Cycle of Servlet, Types of Servlet, Handling HTTP Requests and Responses using GET and POST methods, Deployment Descriptors, Multithreading in Servlets, Session Tracking, Using Cookies, Servlet Filters- Implementing the Filter Interface, The MVC Architecture.

Unit 5: Java Server Pages **[12]**

Introduction to JSP- JSP Development, Basic JSP Lifecycle, JSP Elements, Creating and Deploying a JSP Web Application, Using Implicit Objects- The Request Object, The Response Object, The Out Object-The Session Object, The config Object, The Exception Object, The Application Object, Using Standard Actions and Implicit Objects in JSP Pages, Translation and Compilation, Handling Error and Exceptions-Dealing with Exceptions through the Page Directive, Dealing with Exceptions in the Deployment Descriptor, Adding Exception Handling in JSP Pages, Including and Forwarding from JSP Pages- Expression Language, Custom Actions and Tag Handlers JSP Standard Tag Library (JSTL).

Reference books:

- Java the complete Reference by Herbert Schildt
- Java Servlet Programming by Jasan Hunter
- Beginning Java EE5 from Novice to Professionals by K. Makhar & C. Zelenk
- Java Server Programming by Bayross & Shah
- Thinking in java by Brucel

Type: SEC 3

Course Title: Advanced Python Programming

(Paper Code: Paper XIII)

Course Objectives:

1. To learn Multithreaded Programming.
2. To learn GUI programming using different types of python modules.
3. To study database programming using MySQL.
4. To study Web server programming using CGI and XML.
5. To study Statistical Data analysis and Generating Reports using pandas and matplotlib modules.
6. To learn socket programming concept using Networking.

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Design, Create, Build, and Debug Python applications.
2. Explore Integrated Development Environment (IDE).

Unit 1: Multithreading **[6]**

Understanding threads, Difference between Process and a Thread, Creating Threads, Thread Synchronization, Deadlock of Threads, Avoiding Deadlock in a Program.

Unit 2: Networking in Python **[5]**

Introduction to Sockets Programming, Server Socket Methods, Client Socket Methods, IP Address, URL, TCP/IP Server, TCP/IP Client, Sending E-mail Application.

Unit 3: GUI Programming: **[11]**

Introduction, Advantages of GUI, Introduction to GUI library, Root Window, Working with Containers: Frame, Canvas Layout Management, Events and Bindings, Font, Colors, drawing on Canvas (line, oval, rectangle, etc.) Widgets: Label, Button, Checkbutton, Entry, Listbox, Message, Radiobutton, Text, Spinbox, Scrollbar, Menu etc. Writing Python Programs for GUI applications.

Unit 4: Database Connectivity using MySQL [4]

Installation of MySQL Database Software, Installing MySQL Connector, Steps for Database Connectivity, Working with MySQL Database: Inserting, Retrieving, Deleting and updating the data working with Stored Procedure.

Unit 5: Introduction to CGI Programming and XML [10]

Introduction to CGI, Architecture of CGI, Web Server Configuration, Http Header, CGI Environment Variables, GET and POST Methods, File Upload, Handling Cookies, Validation and Authentication, Accessing and Managing Users, Introduction to XML, XML Parser Architecture and API's, Parsing XML with SAX API's, Parsing XML with DOM API's.

Unit 6: Python for Data Analysis [8]

Use of pandas module, Install and import module, Creating Series and DataFrame, Updating Series and DataFrame, Exporting and importing data- Excel and MySQL, Introduction to plotting- use of matplotlib, Install and import matplotlib, statistical graphics using matplotlib-Univariate, Bivariate and Multivariate data, Pandas objects- Histograms, Density plot, Scatter plot, Hexbin plot, Boxplot

Reference Books:

- Advanced Python Programming- By Richard Ozer, 2017
- Core python Programming- Dr. R Nageswara Rao
- Pandas for Everyone Python Data Analysis- By Daniel Y. Chen · 2017
- Expert Python Programming,: Become a master in Python-By Michał Jaworski, Tarek Ziade
- MySQL for Python: Database Access Made Easy- A. Lukaszewski

Type: DSE 1 B

Course Title: System Security

(Paper Code: Paper XIV)

Course Objectives:

1. To learn cryptographic tools.
2. To learn security issues regarding user Authentication.
3. To understand the various access control mechanisms.
4. To learn various types of malicious softwares and Denial-of-Service attacks.

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications.
2. Gain familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath.
3. Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.
4. Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.

Unit 1: Cryptographic Tools **[6]**

Confidentiality with Symmetric Encryption, Message Authentication and Hash Functions, Public-Key Encryption, Digital Signatures and Key Management, Random and Pseudorandom Numbers, Practical Application: Encryption of Stored Data.

Unit 2: User Authentication **[6]**

Means of Authentication, Password-Based Authentication, Token-Based Authentication, Biometric Authentication, Remote User Authentication, Security Issues for User Authentication, Practical Application: An Iris Biometric System, Case Study: Security Problems for ATM Systems.

Unit 3: Access Control **[8]**

Access Control Principles, Subjects, Objects, and Access Rights, Discretionary Access Control, Example: UNIX File Access Control, Role - Based Access Control, Case Study: RBAC System for a Bank.

Unit 4: Database Security **[6]**

The Need for Database Security, Database Management Systems, Relational Databases, Database Access Control, Inference, Statistical Databases, Database Encryption, Cloud Security

Unit 5: Malicious Software **[10]**

Types of Malicious Software (Malware), Propagation– Infected Content–Viruses, Propagation–Vulnerability Exploit–Worms, Propagation–Social Engineering–SPAM E-mail, Trojans, Payload–System Corruption, Payload–Attack Agent–Zombie, Bots, Payload–Information Theft– Keyloggers, Phishing, Spyware, Payload–Stealthing–Backdoors, Rootkits,, Countermeasures

Unit 6: Denial-of-Service Attacks **[8]**

Denial-of-Service Attacks, Flooding Attacks, Distributed Denial-of-Service Attacks, Application-Based Bandwidth Attacks, Reflector and Amplifier Attacks, Defenses Against Denial -of-Service Attacks, Responding to a Denial-of-Service Attack.

Reference Books:

- M. Stamp, “Information Security: Principles and Practice,” 2 st Edition, Wiley, ISBN: 0470626399, 2011.
- M. E. Whitman and H. J. Mattord, “Principles of Information Security,” 4 st Edition, Course Technology, ISBN: 1111138214, 2011.
- M. Bishop, “Computer Security: Art and Science,” Addison Wesley, ISBN: 0 -201-44099-7, 2002.
- G. McGraw, “Software Security: Building Security In,” Addison Wesley, ISBN: 0321356705, 2006.

Type: DSE 2 B

Course Title: Compiler Construction

(Paper Code: Paper XV)

Course Objectives:

1. To learn the process of translating a modern high-level language to executable code.
2. To learn an understanding of the fundamental principles in compiler design and to provide the skills needed for building compilers for various situations that one may encounter in a career in Computer Science.
3. To develop an awareness of the function and complexity of modern compilers.
4. To apply the code generation algorithms to get the machine code for the optimized code.
5. To represent the target code in any one of the code formats
6. To understand the machine dependent code
7. To draw the flow graph for the intermediate codes.
8. To apply the optimization techniques to have a better code for code generation

Course Outcomes:

1. To gives you with both theoretical and practical knowledge that is crucial in order to implement a programming language.
2. It gives you a new level of understanding of a language in order to make better use of the language (optimization is just one example).

Unit 1: Introduction to compiling **[4]**

Compiler, self-compiler, cross compiler, boot strapping , phases of compiler, compiler construction tools, a simple one pass, two pass and multi pass compiler, factor affecting pass structure of compiler.

Unit 2: Lexical Analysis **[6]**

Role of lexical analyzer, input buffering, specification and recognition of tokens, finite automata implications, designing a lexical analyzer generator

Unit 3: Syntax Analysis **[8]**

Role of Parser, writing grammars for context free environments, top down parsing, recursive descent and predictive parsers (LL), Bottom-up parser, Operator precedence Parsing, LR, SLR and LALR parsers.

Unit 4: Syntax Directed Translation **[6]**

Syntax directed definitions, construction of syntax tree, bottom-up evaluation of S-attributed definitions, L-attributed definitions, Top-down translation and Bottom – up evaluation of inherited attributes, analysis of syntax directed definitions.

Unit 5: Run time environments **[4]**

Source language issues, storage organization and location strategies, parameter passing, symbol table organization and generation, dynamic storage allocation.

Unit 6: Intermediate code generation **[4]**

Intermediate languages, declarations, assignments statements and Boolean expressions, case statements, back patching, procedure calls.

Unit 7: Code generation **[6]**

Issues in design of a code generator and target machine, run time storage management ,basic blocks and flow graphs, next use information and simple code generator, issue of register allocation, assignment and basic blocs, code generation from DAG and the dynamic code generation algorithm.

Unit 8: Code optimization **[6]**

Source of optimization, peephole optimization and basic blocks loop in flow graphs, data flow analysis and equations, code improving transformation and aliases, data flow analysis and algorithms, symbolic debugging of optimized code.

Reference Books:

- Compilers Priciple, Techniques, Tools by Aho, Lam, Sethi and Ulman
- Compiler Design by Wihelm, Mauer
- Compiler Design : Theory, Tools and Examples by Bergamann

Type: DSE 3 B

Course Title: Internet Programming using ASP.Net

(Paper Code: Paper XVI)

Course Objectives:

1. To understand the Asp.net architecture.
2. To learn the various web server controls and validation controls.
3. To know the concept of master page, themes and site navigation.
4. To understand the state management techniques and its types.
5. To learn Ajax client and server site technology.
6. To know the web services
7. To learn the different data transactions using ADO.NET

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Create, Design, Debug and Deploy Web applications.
2. Explore Integrated Development Environment (IDE).

Unit 1: Introduction of Asp.Net

Evaluation of Asp.Net, Fundamentals of ASP.NET, Understanding architecture ASP.NET, Compilation Technique of ASP.Net, Application Location, Web Page and Web Site life cycle, ASP.Net Page Structure, Page Directives, Self-page and Cross page posting, Postback and ViewState concepts, Application Folders.

Unit 2: Web Server Control

Creating ASP.NET Pages – Web Forms, Working with web controls – Standard, control group, Rich Controls, Different type of List controls, FileUpload, AdRotator, MultiView, Calendar, Create Web User Control

Unit 3: Validation controls

Introduction of validation, Types of validation, Validation Controls, Validation Groups.

Unit 4: Master Pages and Themes

Need of Master Pages, Basics of master pages, Creating Master and Content pages, Programmatically assign master pages, Nested Master pages, Event ordering of master pages, Basic Themes and Skins, Creating and Using Themes, Defining multiple skins, Programmatically working with themes.

Unit 5: Site Navigation

Site Navigation technique, SiteMapPath, TreeView and Menu Control, Nesting sitemap file, Attach XML file to treeview and menu.

Unit 6: State Management

Introduction of state management, technique, Types of State Management technique-Client side and server side State Management.

Unit 7: AJAX

What is AJAX and need for AJAX, Client side and server side AJAX, Implementing AJAX with JavaScript, Using ASP.NET Ajax Control toolkit, Working with AJAX's Server side controls, ScriptManager, ScriptMangerProxy, Updatepanel, UpdateProgress, Timer.

Unit 8: Web Services

What is Web Service? Understanding SOAP, WSDL, Proxy etc. Creating Web services, How to consume web services, to build an WebService application and Client

Unit 9: Storing and Retrieving Data with ADO.NET

Accessing Data with ADO.NET, Using Data Sets on Web Forms, Processing Transactions, Working with DML commands.

Reference Books:

- "Professional ASP.Net"-Evjen, Sivkumar, Wrox Press.
- "The Complete Reference: Asp.Net"-MacDonald, Tata McGraw Hill.
- "The Complete Reference: Ajax"- Powell, Tata McGraw Hill.
- "Asp.Net Step by step"- George Shephera-Microsoft Press
- Complete reference crystal reports-Geogre Peak.

Type: DSE 4 B

Course Title: Angular JS

(Paper Code: Paper XVII)

Course objectives:

1. Reduce the amount of code you write to build rich user interface applications.
2. Increase the reliability and maintainability of UI by using data binding.
3. Retrieve data from back-end server, manipulate it and display it with ease.
4. Modularize your code with the custom services and directives.
5. Providing two way binding of data.
6. Create Single Page Applications (SPA).

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Create, Design, Debug and Deploy Web applications.
2. Explore Integrated Development Environment (IDE).

Unit 1: Introduction to JavaScript **[8]**

Including scripts on a page, adding statements or expressions, comments, functions, parameters and return values, primitive types, JavaScript operators, Equality Vs. Identity, pre, post increment, Reading and modifying objects properties, adding methods to objects, Control flow statements, working with arrays, Error handling mechanisms using try/catch/finally, throwing our own exceptions.

Unit 2: Basics of Angular JS **[9]**

Introduction to Angular JS, Features of Angular JS, MVC Architecture, Setting up the Environment, First Simple Application, Working with Directives- Directive lifecycle, Using Angular JS built-in directives, Core Directives, Conditional Directives, Style Directives, Mouse and Keyboard Events Directives, Matching directives, creating a custom directive. Expressions and Data Biding- Number and String Expressions, Object Binding and Expressions, Working with Arrays, Forgiving Behavior, Understanding Data binding.

Unit 3: Controllers [6]

Understanding Controllers, Programming Controllers and scope object, Adding Behavior to a Scope Object, Passing Parameters to the Methods, Array as members in Controller Scope, Nested Controllers and Scope Inheritance, Multiple Controllers and their scopes.

Unit 4: Filters and Modules [6]

Filters: Built-in filters, Uppercase and Lowercase Filters, Currency and Number Formatting Filters, OrederBy Filter, Filter Filter, Creating custom filters. AngularJS Modules- Introduction to Angular JS Module, Module Loading and Dependencies, Recommended Setup of Application, Creation vs Retrieval.

Unit 5: Forms [10]

Angular JS Forms: Working with Angular Forms, Model binding, Understanding Data Binding, Binding controls to data, Form controller, Validating Angular Forms, Form events, Updating models with a twist, $error object, Scope-What is scope, Scope lifecycle, Two way data binding, Scope inheritance, Scope and controllers, Scope and directives, $apply and $watch, Rootscope, Scope broadcasting, Scope events.

Unit 6: Services and Ajax in Angular JS [5]

Understanding Services, Developing Creating Services, Using a Service, Injecting Dependencies in a Service.
$http Service, $q Service, Ajax Implementation using $http and $q Service

Reference Books:

- Beginning AngularJS- By Andrew Grant- 2014
- Professional AngularJS by Diego Netto and Valeri Karpov-Wrox press
- Learning AngularJS by Brad Dayley- Addison-Wesley Professiona
- AngularJS by Brad Green and Shyam Seshadri- O'Reilly

Type: SEC 4

Course Title: Mobile Application Development

(Paper Code: Paper XVIII)

Course Objectives:

1. To understand Android platform and its architecture.
2. To learn about mobile devices types and different modern mobile operating systems.
3. To learn activity creation and Android User Interface designing.
4. To learn basics of Intent, Broadcast and Internet services.
5. To learn about different wireless mobile data transmission standards.
6. To understand and learn how to integrate basic phone features, multimedia, camera and Location based services in Android Application.
7. To learn about different systems for mobile application development, deployment and distribution in Mobile market place (Android, IOS).
8. To understand and carry out functional test strategies for mobile applications.

Course Outcomes: Upon successful completion of this course, students will be able to-

1. Create, Design, Debug and Deploy Android applications.
2. Explore Integrated Development Environment (IDE).

Unit 1: Introduction

What is Android, Android Versions and its Feature Set, Various Android Devices on the Market, Android Market Application Store, Android Development Environment, System Requirements, Android SDK, Installing Java, and ADT bundle - Eclipse Integrated Development Environment (IDE), Creating Android Virtual Devices (AVDs).

Unit 2: Android Architecture Overview and Application

Android Software Stack, The Linux Kernel, Android Runtime - Dalvik Virtual Machine, Android Runtime – Core Libraries, Dalvik VM Specific Libraries, Java Interoperability Libraries, Android Libraries, Application Framework, Creating a New Android Project, Defining the Project Name and SDK Settings, Project Configuration Settings, Configuring the Launcher Icon, Creating an Activity, Running the Application in the AVD, Stopping a Running Application, Modifying the Example Application, Reviewing the Layout and Resource Files.

Unit 3: Android Software Development Platform and Framework

Understanding Java SE and the Dalvik Virtual Machine, The Directory Structure of an Android Project, Common Default Resources Folders, The Values Folder, Leveraging Android XML, Screen Sizes , Launching Mobile Application: The AndroidManifest.xml File, Android Application Components, Android Activities: Defining the UI, Android Service s: Processing in the Background, Broadcast Receivers: Announcements and Notifications Content Providers: Data Management, Android Intent Objects: Messaging for Components, Android Manifest XML: Declaring Your Components.

Unit 4: Understanding Android User Interfaces, Views and Layouts

Designing for Different Android Devices, Views and View Groups, Android Layout Managers, The View Hierarchy, Designing an Android User Interface using the Graphical Layout Tool, Displaying Text with Text View, Retrieving Data from Users, Using Buttons, Check Boxes and Radio Groups, Getting Dates and Times from Users, Using Indicators to Display Data to Users, Adjusting Progress with Seek Bar, Working with Menus using views, Gallery, Image Switcher, Grid View, and Image View views to display images, Creating Animation.

Unit 5: Databases, Intents, Location-based Services

Saving and Loading Files, SQLite Databases, Android Database Design, Exposing Access to a Data Source through a Content Provider, Content Provider Registration, Native Content Providers Intents and Intent Filters: Intent Overview, Implicit Intents, Creating the Implicit Intent Example Project, Explicit Intents, Creating the Explicit Intent Example Application, Intents with Activities, Intents with Broadcast Receivers.

Unit 6: Application Development

Sending SMS Messages Programmatically, Getting Feedback after Sending the Message, Sending SMS Messages Using Intent Receiving, Sending email, Introduction to location-based service, Configuring the Android Emulator for Location -Based Services, Geocoding and Map-Based Activities Multimedia: Audio, Video, Camera: Playing Audio and Video, Recording Audio and Video, Using the Camera to Take and Process Pictures.

Reference Books

- Android Programming Unleashed (1st Edition) by Harwani.
- Beginning Mobile Application Development in the Cloud (2011), Richard Rodger.
- Learn Android App Development by Wallace Jackson.
- Professional Android Application Development by Reto Meier.

Practical Assignments on DSE 2 A and DSE 2 B

Tool: JFLAP

Practical Assignments on DSE 2 A: (Theory of Computer Science)

The students are expected to understand JFLAP tools and designing suitable automata to recognize the following and test the output as string accepted or rejected.

1. Design a DFA which accept string is start with a over ∑={a,b}.
2. Design a DFA which accept number is even or odd.
3. Design a DFA which accept string either ending with ab or bc over ∑={a,b,c}.
4. Design a DFA which accept string does not having abc as substring over ∑={a,b,c}.
5. Design DFA which accept strings length is odd over ∑={0,1}.
6. Design Moore machine for finding binary is even or odd.
7. Design a Mealy machine for 1’s complement of binary number.
8. Design Mealy machine for accepting 2’s complement of binary number.
9. Removal unit production of following grammar;

I)	A -> B	II)	S -> A
	B -> a		S -> bb
	B -> b		A -> B
			A -> b
			B -> a
			B -> S

10. Removal ε-production of following grammar;

I)	S -> aSa	II)	S -> a
	S -> bSb		S -> Xb
	S -> ε		S -> aYa
			X -> Y
			X -> ε
			Y -> b
			Y -> ε

11. Convert following Context Free Grammar(CFG) to Chomsky Normal Form (CNF);

I) S -> ABAB	II) S -> ABA
A -> Aa	A -> aA
A -> a	A -> ε
B -> b	B -> bB
	B -> ε

12. Design a PDA for accepting palindrome string over ∑= {0,1}.

13. Design a PDA to check whether a given string over {a,b} ends in abb.

14. Design TM for L= {$a^n b^n$| n>1}.

15. Construct Turing machine for copy string over ∑={a,b}.

16. Construct Turing Machine that recognize the language:

17. L= {x ϵ {0,1} *| x ends in 00}.

Practical Assignments on DSE 2 B: (Compiler Construction)

1) Convert following Grammar to LL Grammar;

I) S -> ABcC A -> aA A -> ϵ B -> bbB B -> ϵ C -> BA	II) S -> ABcC A -> aA A -> ϵ B -> bbB B -> ϵ C -> BA

2) Build LL(1) parse table for following Grammar and find out LL(1) or not;

I. S -> A A -> aaA A -> b	II. E -> E+T T -> T*F F -> (E) F - > a
III. S -> aA S -> a A -> a	IV. S -> AS S -> a A -> SA A -> b
V. S -> BAc A -> aA A -> a B -> AB B -> bB B -> d	

3) Build SLR(1) parse table for following Grammar and find out LL(1) or not;

I.	S -> A S -> B A -> aA A -> b B -> dB B - > b	II.	E -> E+T T -> T*F F -> (E) F - > a
III.	S -> AA S -> aA A -> b	IV.	S -> A A -> aaA A -> b
V.	S -> Ab A -> aA A -> ab A -> ϵ		

Practical Assignments on DSE 3 A and DSE 3 B

Practical Assignments on DSE 3 A: (Visual Programming)

1. Write a program to check entered number is even or odd.
2. Write a program to get number and display sum of digits.
3. Write a program to check whether entered year is leap year or not.
4. Write a program to display date in various formats.
5. Write a program to illustrate the Use of Access Specifiers.
6. Write a Program to create sealed class.
7. Write a Program to perform boxing and unboxing operation.
8. Write a Program to demonstrate multilevel inheritance.
9. Write a Program to demonstrate single level inheritance.
10. Write a Program to demonstrate multilevel inheritance with virtual methods.
11. Write a Program to get lower bound and upper bound of an array.
12. Write a Program to demonstrate jagged array.
13. Write a Program to find Minimum and Maximum of numbers.
14. Write a Program to search elements of an array.
15. Write a Program to copy a section of one array to another.
16. Write a Program to demonstrate abstract properties.
17. Write a Program to implement delegates.
18. Write a Program to combine two delegates.
19. Write a Program to implement multicast delegate.
20. Write a Program to demonstrate DivideByZero Exception.
21. Write a Program to demonstrate Multiple exceptions.
22. Write a Program to create a file.
23. Write a Program to Read the Contents of File.
24. Write a Program to Create Directory.
25. Write a Program to implement BinaryReader.
26. Write a Program to Read Line from File until end of file is reached.
27. Write a Program to Design user interface using all windows controls.
28. Write a Program to design MDI application. 29. WAP to demonstrate ADO.NET.
29. Write a Program to demonstrate Insert, Update and Delete Statements.

Practical Assignments on DSE 3 B: (Internet Programming using ASP.Net)

1. Design web page for student admission which uses Label, TextBox, RadioButton, CheckBox, ListClass, ButtonClass, Calendar, Image, FileUpload etc. controls.
2. Design scientific calculator.
3. Design web page which demonstrate command name property.
4. Design web page which demonstrate which code is execute at first either server side or client side.
5. Design web page for Self Page Posting and Cross Page Posting.
6. Design web page which demonstrate App_code using class library. Class library contains methods which checks odd, even, prime, Armstrong, Palindrome, Strong and Magic number.
7. Design web page which demonstrate App_GlobalResources and App_LocalResources.
8. Design web page which demonstrate page lifecycle and website lifecycle.
9. Design simple application which displays selected checkboxes and radio button.
10. Design a web page for image mapping using static and dynamic method.
11. Demonstrate all methods of insertion of item in list class.
12. Design web page which displays all system fonts, system colors, font size in List Class. Display text message according to the selected font, size and color.
13. Display Current Year calendar. This calendar shows all holidays in Red color with information.
14. Display selected date in at least 10 different formats.
15. Designs XML file which shows College-Stream-Department-Staff-name-quali-exp-subject.
16. Display at least 10 different advertisements.
17. Design a web page for Wizard and MultiView control.
18. Design a web page which displays 10 textbox controls by using control array method.
19. Design web page which uses all validation controls with validation group property.
20. Design Nested master pages using themes.
21. Design web page which demonstrate working of DML Queries.

Practical Assignments on DSE 4 A and DSE 4 B

Practical Assignments on DSE 4 A: (Advanced Java)

1. Write a java socket programming in which client sends a text and server receives it.
2. Write a program to demonstrate URL class.
3. Write a program to demonstrate InetAddress class.
4. Write a program to demonstrate use of Datagram Socket.
5. Write a program to create Student registration form using Swing Component.
6. Write a program to demonstrate JTabbedPane, JScrollPane and JTree Component.
7. Write a program for inserting data into table using PreparedStatement.
8. Write a program for updating data into table using PreparedStatement.
9. Write a program for deleting data into table using PreparedStatement.
10. Write a program to demonstrate callable statement.
11. Write a servlet based program to display “Hello!” message on browser.
12. Write a program that reads parameters from servlet and display it.
13. Write a servlet to handle Http GET Request.
14. Write a servlet to handle Http POST Request.
15. Write a servlet based program that will add the cookie and display all the cookies stored.
16. Write a servlet based program that will display the last login date and time of a user.
17. Write a servlet based program the will display how many times a user visited to the page.
18. Write a program that implements Filter interface and servlet filter mapping in web.xml.
19. Write a servlet based program for login form validation.
20. Create a simple HTML form, Example.html, which allows a user to enter a number. Example.html submits the number captured to the web server when the user clicks on the form’s Submit button. The web server executes Example.jsp code spec and responds with an Html page with Hello World! Repeated as many times as specified.
21. Create a jsp page that will display current data and time.
22. Create a user interface that accepts data using an Html form and display the same on page submission.

23. Create an html page [user interface] for capturing Book information. This form will accept data from the user and display the same upon page submission.
24. Create an html page [user interface] for capturing Book information. This form will accept data from the user and display the same upon page submission.
25. Create the following files to demonstrate the <jsp:forward> action element:
26. Write an application to demonstrate the <jsp:plugin> action element.
27. Create a registration form using action elements, perform the following steps: register.jsp- This file holds the actual registration form interface. The data captured here is submitted to the process.jsp file for further processing. Process.jsp- This file validates the form contents i.e. data captured using a method of FormBean.java and if data is found valid, then control is forwarded to success.jsp file. If data if found invalid then control is shifted back to register.jsp file. Success.jsp- This file retrieves the captured and validated data using bean and displays the same. FormBean.java- This file does the actual validation.
28. Create a jsp page that uses the implicit object.
29. Create a custom tag and use it in a jsp page.

Practical Assignments on DSE 4 B: (Angular JS)

1. Write an AngularJS simple Hello World! Program.
2. Write an angularJSprogram that display your Roll No, Name and Class of student.
3. Write an angularJS program which demonstrate that one-way data binding and two-way data binding.
4. Write an angularJS program which demonstrate ng-cut, ng-copy, ng-paste directive.
5. Write an angular JS program which demonstrate different directive related to keyboard.
6. Write an angularJSprogram which demonstrate conditional directive.
7. Write an angularJS program for creating custom directive which display employee id and name.
8. Write an angularJS program which demonstrate all types of expressions 1) Number expression 2) String expression 3) Object expression 4) Array expression
9. Write an angularJS program to demonstrate use of nested controller.
10. Write an angularJS program to demonstrate multiple controller
11. Write an angularJS program to demonstrate json filter
12. Write an angularJS program to demonstrate custom filter
13. Write an angularJS program to design simple single page application.
14. Write an angularJS program to create Custom validation in.

Course Title: Project Work

Instructions:

1. Team size for major project not exceed than two students.
2. Real time and live project followed by Presentation and Viva-Voce.

Equivalent Subject for Old SyllabusB.Sc. (Entire Computer Science) - III (Semester –V and VI)

Semester-V		
Sr. No.	**Name of the Old Paper (w.e.f.2018-19)**	**Name of the New Paper (w.e.f.2021-2022)**
1	English	English (Business English) Paper-II Part-A (Sem-V)
2	Data Communication and Networking - I	Data Communication and Networking(Sem-V)
3	Theory of Computer Science	Theory of Computer Science(Sem-V)
4	Visual Programming - I	**No Equivalence**
5	Web technology and E-commerce –I	**No Equivalence**
6	Python – I	**No Equivalence**
Semester-VI		
Sr. No.	**Name of the Old Paper (w.e.f.2018-19)**	**Name of the New Paper (w.e.f.2021-2022)**
1	English	English (Business English) Paper-II Part-B (Sem-VI)
2	Data Communication and Networking – II	System Security(Sem-VI)
3	Advanced Java	Advanced Java (Sem-V)
4	Visual Programming -II	Visual Programming (Sem-V)
5	Web technology and E-commerce –II	Internet Programming using ASP.Net(Sem-VI)
6	Python - II	Advanced Python Programming (Sem-V)