Appendix-A

Master of Science (M.Sc. Computer Software) Full Time Two Years Degree Programme :-

1) A Student shall have to be admitted every year in the respective Institute/ College/University Department for completion of an academic year of this two year Degree program.
2) The M.Sc. Degree shall consist of four semesters i.e. Semester I & II in the first academic year, Semester III & IV in the second academic year.
3) Student has to complete all four Semesters successfully for the award of Degree of Master of Science and fulfill conditions as per Ordinance No. 19.
4) Every Semester of M.Sc. Computer Software Program shall be of at least 90 teaching days in a semester and shall be of at least 180 teaching days in an academic year.
5) The Examinations shall consist of the subjects as indicated in the Scheme of Examinations as per **Appendix – 'A1 to A4'.**

6) The Semester wise structure of the program is as follows.

Sant Gadge Baba Amravati University Amravati

Scheme of Teaching, Learning & Examination leading to the Degree Master of Science (Computer Software)

(Two Years- Four Semesters Degree Programme- C.B.C.S)

(M. Sc. Part-I) Semester- I

S. N.	Subject Code	Type	Subject	Teaching & Learning Scheme							Duration Of Exam Hours	Examination & Evaluation Scheme						
				Teaching Period Per Week				Credits				Maximum Marks					Minimum Passing	
												Theory		Practical		Total Marks		
				L	T	P	Total	L/ T	Practical	Total		Theory+ MCQ External	Theory Internal	Intern al	External		Mark s	Grade
	Core Subject																	
1	1MCS1	DSC1	1 Computer System Design	4	-	-	4	4	-	4	3	80	20			100	40	P
2	1MCS2	DSC2	2 Data Structure with OOP	4	-	-	4	4	-	4	3	80	20			100	40	P
3	1MCS3	DSC3	3 Data Base Management Technologies	4	-	-	4	4	-	4	3	80	20			100	40	P
4	1MCS4	DSC4	4 Computer Network & Wireless Technology	4	-	-	4	4	-	4	3	80	20			100	40	P
	Skill																	
5	1MCS5	SEC1	4-Advanced Java/ NS2/ tools	-	2	2	4	4	-	4	3	-	-	25	25	50	25	P
	Elective																	
6.	1MCS6(1)	DSE1	(1) Discrete Mathematical Structure	4	-	-	4	4	-	4	3	80	20			100	40	P
7.	1MCS6(2)	DSE2	(2) Entrepreneurship Development															
8.	1MCS6(3)	DSE3	(3)Research Methodology															
9.	1MCS6(4)	DSE4	(4)Management Information System															
10.	1MCS6(5)	DSE5	(5)Data Science and Analytics															
	Laboratories																	
11	1MCS7	Lab-I	1,2 - Programming(C/C++/Java/ALP)	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
12	1MCS8	Lab-II	3-SQL/ DBMS tools, MSsql, My Sql	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
	Internship																	
13	1MCS9		**#Internship/Field Work/Work Experience@**															
			TOTAL	20	2	10	32											
	Open Elective																	
14	1MCS10	OE1	**Open Elective (OE) /GIC/Open skill/MOOC***	-	2	-	2	-	1	1	-			25	25	50	25	P
			TOTAL	20	4	10	34	24	5	29								
		GIC1	**User Experience Deign**															
		GIC2	**Effective Email Communication**															

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Software) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC (Optional) can be studied during semester I to IV

Sant Gadge Baba Amravati University Amravati

Scheme of Teaching, Learning & Examination leading to the Degree Master of Science (Computer Software)

(Two Years- Four Semesters Degree Programme- C.B.C.S)

(M. Sc. Part-I) Semester- II

S. N.	Subject Code	Type	Subject	Teaching & Learning Scheme							Duration Of Exam Hours	Examination & Evaluation Scheme						
				Teaching Period Per Week				Credits				Maximum Marks					Minimum Passing	
												Theory		Practical		Total Marks		
				L	T	P	Total	L/ T	Practical	Total		Theory+ MCQ External	Theory Internal	Intern al	External		Mark s	Grade
	Core Subject																	
1	2MCS1	DSC1	1 Operating System Algorithms	4	-	-	4	4	-	4	3	80	20			100	40	P
2	2MCS2	DSC2	2 Graphics Application programming	4	-	-	4	4	-	4	3	80	20			100	40	P
3	2MCS3	DSC3	3 Software Engineering	4	-	-	4	4	-	4	3	80	20			100	40	P
4	2MCS4	DSC4	4 Data Mining and Data Warehousing	4	-	-	4	4	-	4	3	80	20			100	40	P
	Skill																	
5	2MCSW5	SEC2	1 -OS (Windows / Android /Linux)	-	2	2	4	4	-	4	3	-	-	25	25	50	25	P
	Elective																	
6.	2MCS6(1)	DSE1	(1)Theory of Computation	4	-	-	4	4	-	4	3	80	20			100	40	P
7.	2MCS6(2)	DSE2	(2)Computer System Architecture															
8,	2MCS6(3)	DSE3	(3)Enterprise Resource Management															
9.	2MCS6(4)	DSE4	(4)Mobile Computing															
10.	2MCS6(5)	DSE5	(5)Compiler Construction															
	Laboratories																	
11	2MCS7	Lab-III	2- Graphics programming and tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
12	2MCS8	Lab-IV	3,4 -SE tools/ DM tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
	Internship																	
13	2MCS9		**#Internship/Field Work/Work Experience@**															
			TOTAL	20	2	10	32											
	Open Elective																	
14	2MCS10	OE2	**Open Elective (OE) /GIC/Open skill/MOOC***	-	2	-	2	-	1	1	-			25	25	50	25	P
			TOTAL	20	4	10	34	24	5	29								
		GIC3	**Web Page Design Techniques**															
		GIC4	**Automation With Robotics**															

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Software) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC (Optional) can be studied during semester I to IV

Appendix A3

Sant Gadge Baba Amravati University Amravati

Scheme of Teaching, Learning & Examination leading to the Degree Master of Science (Computer Software)

(Two Years- Four Semesters Degree Programme- C.B.C.S)

(M. Sc. Part-II) Semester- III

S. N.	Subject Code	Type	Subject	Teaching & Learning Scheme							Duration Of Exam Hours	Examination & Evaluation Scheme						
				Teaching Period Per Week				Credits				Maximum Marks					Minimum Passing	
												Theory		Practical		Total Marks		
				L	T	P	Total	L/ T	Practical	Total		Theory+ MCQ External	Theory Internal	Intern al	External		Mark s	Grade
	Core Subject																	
1	3MCSW1	DSC1	1 Open Source Software Technologies	4	-	-	4	4	-	4	3	80	20			100	40	P
2	3MCSW2	DSC2	2 Web Development and CMS	4	-	-	4	4	-	4	3	80	20			100	40	P
3	3MCSW3	DSC3	3 Applied Machine Learning	4	-	-	4	4	-	4	3	80	20			100	40	P
4	3MCSW4	DSC4	4 Distributed Systems	4	-	-	4	4	-	4	3	80	20			100	40	P
	Skill																	
5	3MCSW5	SEC3	1 Programming with .Net Technology	-	2	2	4	4	-	4	3	-	-	25	25	50	25	P
	Elective																	
6.	3MCSW6(1)	DSE1	(1) Optimization Techniques	4	-	-	4	4	-	4	3	80	20			100	40	P
7.	3MCSW6(2)	DSE2	(2) Cloud Computing															
8,	3MCSW6(3)	DSE3	(3)Software Project Management															
9.	3MCSW6(4)	DSE4	(4) Bioinformatics Techniques															
10.	3MCSW6(5)	DSE5	(5) Information System Security															
	Laboratories																	
11	3MCSW7	Lab-V	1,2 Web development tools/CMS tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
12	3MCSW8	Lab-VI	3 Tools for ML/SciLab/ Python	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
	Internship																	
13	3MCSW9		**#Internship/Field Work/Work Experience@**															
			TOTAL	20	2	10	32											
	Open Elective																	
14	3MCSW10	OE3	**Open Elective (OE) /GIC/Open skill/MOOC***	-	2	-	2	-	1	1	-			25	25	50	25	P
			TOTAL	20	4	10	34	24	5	29								

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Software) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC (Optional) can be studied during semester I to IV

Sant Gadge Baba Amravati University Amravati **Appendix A4**

Scheme of Teaching, Learning & Examination leading to the Degree Master of Science (Computer Software)

(Two Years- Four Semesters Degree Programme- C.B.C.S)

(M. Sc. Part-II) Semester- IV

S. N.	Subject Code	Type	Subject	Teaching & Learning Scheme							Duration Of Exam Hours	Examination & Evaluation Scheme						
				Teaching Period Per Week				Credits				Maximum Marks					Minimum Passing	
												Theory		Practical		Total Marks		
				L	T	P	Total	L/ T	Practical	Total		Theory+ MCQ External	Theory Internal	Intern al	External		Mark s	Grade
	Core Subject																	
1	4MCSW1	DSC1	1 Software Testing	4	-	-	4	4	-	4	3	80	20			100	40	P
2	4MCSW2	DSC2	2 Big Data Analytics	4	-	-	4	4	-	4	3	80	20			100	40	P
3	4MCSW3	DSC3	3 Internet of Things(IOT) Technology	4	-	-	4	4	-	4	3	80	20			100	40	P
4	4MCSW4	DSC4	4 Cyber Security and Digital Forensics	4	-	-	4	4	-	4	3	80	20			100	40	P
	Skill																	
5	4MCSW5	SEC4	Android Programming	-	2	2	4	4	-	4	3	-	-	25	25	50	25	P
	Elective																	
6.	4MCSW6(1)	DSE1	(1)Parallel Computing	4	-	-	4	4	-	4	3	80	20			100	40	P
7.	4MCSW6(2)	DSE2	(2)Image Processing															
8,	4MCSW6(3)	DSE3	(3) Block chain Technology															
9.	4MCSW6(4)	DSE4	(4)OOSE															
10.	4MCSW6(5)	DSE5	(5)Robotics& AI															
	Laboratories																	
11	4MCSW7	Lab-VII	1,2 – Testing and Data analytics tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
12	4MCSW8	Lab-VIII	3,4 – IoT tools / Security tools	-	-	4	4	-	2	2	3	-	-	25	25	50	25	P
13	4MCSW9		Seminar	2				1		1				25	25	50	25	P
14	4MCSW10		Project			4			2	2				50	50	100	50	P
	Internship																	
15	4MCS11		#Internship/Field Work/Work Experience@															
			TOTAL	22	2	14	38	25	6	31								
	Open Elective																	
16	4MCS12	OE4	Open Elective (OE) /GIC/Open skill/MOOC*	-	2	-	2	-	1	1	-			25	25	50	25	P
			TOTAL	22			40	25	7	32								

L: Lecture, T: Tutorial, P: Practical

Students may complete their Internship/Field Work/Work Experience in First OR Second OR Third Semester of M. Sc. (Computer Software) according to their convenience; @ denotes Ancillary Credit

Note : Internship /Apprenticeship/Field Work / Work Experience (During vacations of Semester I to Semester III) for duration of minimum 60 hours to maximum 90 hours mandatory to all the students, to be completed during vacations of Semester I to III. This will carry 2 Credits for learning of 60 hours or 3 Credits for learning of 90 hours. Its credits and grades will be reflected in final semester IV credit grade report.

- OEC (Optional) can be studied during semester I to IV

Total Credits: 119

Common Instructions for all the Semesters regarding Choice Based Credits (CBC)/Open Electives (OE) are as under:
The titles of broad activity those can be undertaken by the students in every semester and their respective credits are listed in the table given below. Student has to undertake one or more activities out of these table so as to avail at least 2 credits per semester

The Subjects/Modules Activity to be undertaken by the Student under the Open Electives approved by the Department Institute. The schedule of approval will be declared by the Department/Institute at the beginning of the Semester (1 July) as per details given below:

One Faculty Member will work as a Coordinator for Open Electives for which 01 Hour of Theory period will be considered as a weekly work load against this work. All Coordinators has to do counselling of respective Open electives, do the Students Registration process and allot them to faculty members (will be working as a mentor). All these electives are internally accessed by respective Coordinators & Guides based on Minimum 03 Class Tests/ Final Objective Test/ Demo/ Report Submission/Certificate issued by competent authority Viva Voce and other methods as decided by the Department/Institute.

The Mentor shall conduct Tutorial Classes for Workload counting purpose, it should be noted that: 01 Tutorial hour is equal to 01 Theory Hour. For Tutorial, Batch of Maximum 20 Students will be considered and the Tutorial Batch should not be comprised of Less than 04 Students.

Coordinator shall take care that the students are not repetitively opting for same type of Electives in every Semester.

Summary of conduction of Choice Based Credits (CBC)/Open Electives (OC) Electives for all Semesters

i. Electives Selection Process starts at beginning of the Semester,
ii. Declare the names of Coordinator for Open Electives.
iii. Counselling of Students by Coordinators for selection of Open Electives
iv. Registration of Students by Coordinators under respective Open Electives
v. Allotment of Registered Students to Mentor from Department.
vi. Guidance/Counselling to Students by Mentor throughout the Semester
vii. General Counselling by Coordinators over the Semester, whenever required.
viii. Final Assessment of Students by Coordinators & Mentor for Allotment of Final Credits
ix. Submission of Credits gained by Students to the Head of Department from Coordinators

The Open Elective and Credit Assigned	**Credit**
Successful completion of Online Course of 4 weeks	4
Project activity	4
Seminar Activity	1
Paper/poster presentation	1
Completion of soft skill programme of one week	1
Internship of 30 Hrs	2
Field Visit of 15 Hrs	1
Startup recognized and approved by the department	2

Participation in Unnat Bharat Abhiyan	1 for 15 days, maximum 4
Yoga Meditation camp of 1 week	1
Completion of course/activity of similar credits proposed by the department from among the available courses/activities from other department/faculty in the college/university	4

POs of PG Programme (M.Sc. Computer Software)

The post-graduate students, after completing their study of postgraduate M.Sc. programme, must acquire following characteristics attributes.

- PO-1 Scientific Knowledge, Experimental and Research Skills: The students must be able to demonstrate fundamental and advance concepts in science and apply it in relative specialised areas like research, teaching and government/social/public services.
- PO-2 Communication skills: The students must be able to transmit complex scientific and technical knowledge in clear and concise manner relating to all areas of a subject they studied at PG level.
- PO-3 Critical Thinking & Problem-Solving ability: The students must be able to employ critical thinking and problem-solving skills to find appropriate solutions for the scientific and research problems in the fields related to the subject they studied.
- PO-4 Team leading and working capability: The students must be capable to work independently as well as a team leader or a member either in academic or research institute.
- PO-5 Project Management: The students must be able to identify and mobilize the appropriate resources to manage and complete the undertaken project by observing responsible & ethical conduct and also with laboratory safety and hygiene.
- PO-6 Technological Proficiency to use Modern instrumentations: The students must be capable to handle sophisticated and advanced instruments for their research work.
- PO-7 Environmental and Societal Consciousness: The students must be aware about the environmental & the societal problems and must be capable to use and demonstrate the scientific knowledge to address these problems and to undertake research problems.
- PO-8 Ethics and Human values: The students must be capable to think and behave rationally on the ethical issues they come across at their work place. Also, the students should adopt human values to keep harmony with individuals and with human beings.
- PO-9 National & International perspective: The post graduate students must be able to develop national and international perspective for their career in the chosen field so that they could play a vital role in contributing to national and global development.
- PO-10 Lifelong Learning: The students should adopt lifelong learning to keep pace with emerging trends in academics, research and developing technology.

PSOs for M.Sc. Computer Software programme stated by Computer Science Department

The student graduating with the Degree M.Sc. Computer Software should be able to

- PSO-1 Acquire depth knowledge in computer software and ability to identify, analyse, design, optimize and implement system solutions using appropriate algorithms of varying complexity.
- PSO-2 Basic knowledge in software methods and tools for solving real-life and R&D problems and ability to work in multidisciplinary teams in small- and large-scale projects by utilizing modern software tools and emerging technologies to develop complex products for the societal needs.
- PSO-3 Specialist in Data mining, embedded systems, Mobile computing, distributed computing, Image processing, Pattern recognition, Virtualization techniques and Cloud Computing.
- PSO-4 Competent and complete software professional to meet the requirement of corporate world and Industry standard to provide solutions to industry, society and business.
- PSO-5 Analyst who can apply latest technologies who can analyse and synthesize computing systems through quantitative and qualitative techniques to solve problems in the areas of Information Technology.

Sant Gadge Baba Amravati University, Amravati

Faculty of Science and Technology

Programme: M Sc Computer Science

PROGRAMME OUTCOMES (POs)

Upon completion of the programme successfully, students would be able to

PO1: Problem Analysis
Identify, formulate, review research literature and analyze complex engineering problems in Computer Science and Engineering reaching substantiated conclusions using first principles of mathematics, naturalsciences and engineering sciences.

PO2: Design / Development of Solutions
Design solutions for complex engineering problems and design system components or processes of Computer Science and Engineering that meet the specified needs with appropriate consideration for public health and safety, cultural, societal and environmental considerations.

PO3: Conduct Investigations of Complex Problems
Use research-based knowledge and research methods including design of experiments in Computer Science and Engineering, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO4: Modern tool usage
Create, select and apply appropriate techniques, resources, and modern engineering and IT toolsincluding prediction and modeling to complex activities related to Computer Science with an understanding of the limitations.

PO5: The services to the society
Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice in Computer Science and Engineering.

PO6: Project Management
Demonstrate knowledge and understanding of the computer science and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PROGRAMME SPECIFIC OUTCOMES (PSOs)

Upon completion of the programme successfully, students would be able to

PSO 1: deliver efficient solutions for emerging challenges in the computation domain through continuous learning

PSO2
design, develop, implement computer programs and use knowledge in various domains to identifyresearch gaps and hence to provide solution to new ideas and innovations.

Employability Potential in M.Sc. Computer Science

If you've studied computer science**,** you will have gained many technical and non-technical skills which are highly valued by employers, from leadership to programming. The increasing scope of computer science means you have plenty of choice in a wide variety of highly specialized areas.

Computer technologies are integral to modern life, so you're likely to find your computer science skills in high demand across many different industries. These include financial organizations, management consultancy firms, software houses, communications companies, data warehouses, multinational companies, governmental agencies, universities and hospitals.

As always, it's extremely beneficial to have completed relevant work experience. You should also consider compiling a portfolio of your own independent projects outside of your degree, which could be in the form of programming, moderating online or even building an app. This will demonstrate to employers your interest in the subject and your problem-solving skills, creativity and initiative.

- Application analyst.
- Applications developer.
- Cyber security analyst.
- Data analyst.
- Forensic computer analyst.
- Game designer.
- Games developer.
- Machine learning engineer
- Cyber security analyst

- Data analyst
- Forensic computer analyst
- Game designer
- Games developer
- Machine learning engineer
- Penetration tester
- Software engineer
- Systems analyst
- UX designer
- Web designer & Developer
- Business analyst
- IT sales professional
- IT trainer
- Nanotechnologist
- Network Engineer
- Telecommunications researcher
- Database Manager/ Administrator

Common employers are IT consultancies and IT service providers. However, as most businesses rely on computers to function effectively, there are also opportunities within the IT departments of major organisations in sectors such as:

- Aerospace and Defence
- Agricultural
- Financial Services
- Healthcare
- Manufacturing
- Public And Third Sectors
- Telecommunications
    - Banking
    - E-Commerce
    - Medical
    - Defence
    - Education
    - Communication
    - Automobile Industry
    - Printing Industry
    - Film Industry
    - Entertainment Industry
    - E- Governance
    - Satellite Launching
    - Simulators
    - Research & Development
    - And Lot More...

You can also find opportunities with a range of small to medium-sized enterprises (SMEs).

Another option is to set up your own business, providing IT services such as web design and consultancy.

Computing degrees combine theoretical study and practical projects, teaching you subject-specific skills including:

- Programming Languages
- Hardware Architecture And Construction
- Network Design and Engineering
- Software Engineering
- Multimedia Design
- Software Tools and Packages.

You'll learn how to specify, design and construct computer-based systems, evaluate and recognize potential risks and design creative solutions.

You'll also get more generic skills from your computing degree including:

- Teamwork and Leadership
- Communication
- Problem Solving
- Negotiation

- Time Management and Organisation
- Report Writing
- Numeracy
- Commercial Awareness.

Continuing professional development (CPD) is especially important when you're working with computers as technology and software develops at such a rapid pace.

Core Subjects

Course Code	**3MCSW1**
Course Name	**Open Source Software Technologies**
Total Credits	**4**
Course Outcomes	**Course Outcome:** After completion of this course successfully, students would be able to 1. Understand concepts, strategies, and methodologies related to open source software development. 2. Understand the business, economy, societal and intellectual property issues of open source software. 3. Get familiar with open source software products and development tools currently available in the market. 4. Learn the basics of Python as open source programming language 5. Learn the basics of an open source operating system using Linux. 6. Learn the basic concepts of Open Source Database using MySQL

Units	Contents	Hrs
I	Principles and Open Source Methodologies, What and Why of Open Source, History, Open Source Principles Free Software vs. Open Source vs. Closed Source Open Source Standards, Methodologies, Philosophy Software Freedom, Open Source Software Development Licenses (MPL, GPL, LGPL, etc.), Copyright, Copyleft The ethics of open source	10
II	Introduction to various open source operating systems Linux: History of Linux OS Components of Linux OS/Linux OS Architecture Kernel Mode vs. User Mode Linux File System Processes, Networking User and Group Account Management	10
III	Introduction to various open source programming languages Python: Basic syntax of Python, Keywords and identifiers, Data Types: Numbers, Strings, Tuples, Lists, Dictionaries, Sets Command line arguments, Getting user input Conditional and Looping statements	10
IV	Programming Languages – Python Advanced Functions in Python, Anonymous Functions Python Modules & Packages Generators, Comprehensions and Lambda Expressions Input and Output in Python: Reading and writing text files, Writing Text Files Errors and Exceptions: try….except…else, try-finally clause Multithreading in Python	10
V	Database: Creation and Manipulation Introduction to various open source Databases MySQL: Configuring MySQL database on Linux DDL: CREATE, DROP, ALTER, TRUNCATE, RENAME commands DML: SELECT, INSERT, UPDATE DELETE, MERGE Keys: Unique, Primary, Foreign, Composite Key Working with Strings, Date and Time	10
VI	Database with Python and Open Source Ecosystem Python Connectivity: Using Python SQL Libraries to Connect to a Database SQLite and other Packages, Sorting query results, Working with metadata Ecosystem: Starting an Open Source project Open Source Hardware Open Source design Open source software repositories Social and financial impacts of Open Source Technology	10
	Text Books: 1. Fundamentals of Open Source Software by M. N. Rao, PHI, First Edition 2. Linux in a Nutshell by Ellen Siever, Stephen Figgins, Robert Love, Arnold Robbins, O'Reilly Media, Inc., 6th Edition 3. Learning MySQL: Get a Handle on Your Data by Vinicius Grippa, Sergey Kuzmichev, O'Reilly, 2nd Edition 4. Learning Python by Mark Lutz, O'Reilly Media, Edition: 5th	
	Reference Books: 1. Code Reading: The Open Source Perspective by Diomidis Spinellis Addison-Wesley Professional 2. Producing Open Source Software: How to Run a Successful Free Software Project by Karl Fogel - O'Reilly	

Course Code	**3MCSW2**
Course Name	**Web development and CMS**
Total Credits	**4**
Course Outcomes	**Course Outcome:** After completion of this course successfully, students would be able to **1.** Understand concepts, strategies, and methodologies related to content Management System **2.** Get familiar with CMS products and development tools currently available in the market. **3.** Learn the basics of WordPress **4.** Learn the basics of Joomla **5.** Learn the basic concepts of Drupal

Units	Contents	Hrs
I	Introduction to Content Management System(CMS), Web Content Management System(WCMS), features of WCMS, Types of CMS, **Wordpress**: Introduction, Features, Installation & Configuration, Dashbord, WordPress Settings: General, Writing, Reading, Discussion, Media, Permalink, Plugin, WordPress Categories: Add, Edit, Delete, Arrange, WordPressPosts, WordPressMedia, WordPressPages, Links, Comments.	10
II	**WordPress Plugins:** View, Install, Customize, WordPress Users: User Roles, Add user, Edit user, delete users, WordPress Appearance, Host Transfer, Version Update, Spam Protection, Backup and restore, Optimization, Reset password.	10
III	**Joomla:** Introduction, Installation & configuration, Control Panel, Toolbars, Menus: Create, Modify, Modules: Create, Breadcrumbs, Feed Display, Footer, Search, Random Image, Syndicate, Joomla Gobal Settings: System, Media, Language, Private, Mass E-mailing, Cash Management, User Settings, Debug.	10
IV	**Joomla Advanced:** Template Manager, Customization, Adding and Creating Templates, Customize logo, category management, adding content, formatting content, article metadata, adding banners, adding contacts, adding forums, Plugin manager, Extension manager, Website Backup.	10
V	**Drupal:** Introduction, Installation & configuration, Architecture, Main Menu, Blocks and Regions, Themes and layouts, Pages: Front and static, Create Blog, Create articles, Create Content, Modify Content, Publish Content, Manu Management, Taxonomies, Comments, User Management, Optimization, Site Backup, Site Upgrade .	10
VI	URL Alias, Site Search, Error Handling, Multilingual Content, Triggers and Actions, Social Networking, Internationalization, Extension, Modules: Default Modules, Pane Modules, Book Module, Aggregator Module, Contact Module, Form Module, Poll Module, Site Security Drupal E-Commerce- Setup Shopping Cart, Create Products, Create Categories, Setup Taxes and Discounts, Receives Donations, Set up shipping, Set up payments, Invoice Generations, Email notifications, Order History.	10
	Books: • WordPress:VisualQuickstart Guide By Matt Beck, Jessica Neuman Beck. • Professional wordpress: Design & Development By Brad Williams, David Damstra, Hal Sterm, Wrox Publication • WordPress Complete By HasinHayder, Packt Publishing • Learning Joomla 3 Extension Development-Third Edition - by Tim Plummer • Joomla Programming By Mark Dexter & Louis Landry • Joomla: Beginners Guide By Eric Tiggler • The Official Joomla Book By Jennifer Marriot, ElinWaring • Mastering Drupal 8 – • Beginning Drupal 7- Tom Tomlison, Apress Publishing • Drupal-7 by David Mercer, PACKT Publishing	

Prescribed Syllabus

Semester-III

Course Code	**3MCSW3**
Course Name	**Applied Machine Learning**
Total Credits	**4**
Course Outcomes	**Course Outcome:** After completion of this course successfully, students would be able to **1.** understand the concepts of machine learning. **2.** appreciate supervised and unsupervised learning and their applications. **3.** appreciate the concepts and algorithms of learning.

Units	Contents	Hrs
I	**Introduction**: Definition-Examples of machine learning applications –Well posed learning problems- Designing a learning system- Perspectives and issues Concept learning and general to specific ordering: Inductive learning hypothesis- Concept learning as search – candidate elimination algorithm-inductive bias.	10
II	**Regression and classification** Regression: Linear Regression-Simple-Multiple Decision Tree-Pruning: Introduction – Representation-Algorithm- issues Classification: Support Vector machine – Naïve Bayes-Applications.	10
III	**Clustering and Learning** Clustering: k-Means clustering– adaptive Hierarchical clustering –Applications- Neural network: Perceptron, multilayer network- back propagation- introduction to deep neural network Instance based learning: k -NN– Radial basis functions Case based reasoning-Reinforcement learning -Applications.	10
IV	**Probabilistic graphical models** Graphical Models: Undirected graphical models - Markov Random Fields - Directed Graphical Models -Bayesian Networks - Conditional independence properties - Inference – Learning Generalization - Hidden Markov Models -Conditional random fields(CRFs).	10
V	**Decision Trees:** Decision Trees, Supervised Learning of Univariate Decision Trees, Networks Equivalent to Decision Trees, Overfitting and Evaluation, The Problem of Replicated Subtrees, The Problem of Missing Attributes, Comparisons	10
VI	**Machine learning experiments** Design-Cross validation - Measuring Performance -Hypothesis testing- Assessing Performance -Comparison of algorithms, Datasets-Case study.	10
	Text Books: 1) Tom M. Mitchell, Machine learning, McGraw-Hill, 1997. 2. Ethem Alpaydin, Introduction to Machine Learning, MIT Press, Third Edition, 2014.	
	Reference Books: 1. Nils J. Nilsson ,Introduction to Machine Learning, Robotics Laboratory Department of Computer Science Stanford University, November 3, 1998.	

Course Code	**3MCSW4**
Course Name	**Distributed Systems**
Total Credits	**4**
Course Outcomes	**Course Outcome:** After completion of this course successfully, students would be able to 1. understand Transactions and Concurrency control. 2. understand Security issues 3. Understanding Distributed Multimedia System. 4. design distributed systems for basic level applications.

Units	Contents	Hrs
I	Characterization of Distributed Systems-Introduction, Examples of Distributed systems, Resource sharing and web, challenges, System models -Introduction, Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication, Distributed objects and Remote Invocation-Introduction, Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI.	10
II	Operating System Support- Introduction, OS layer, Protection, Processes and Threads, Communication and Invocation, Operating system architecture, Distributed File Systems-Introduction, File Service architecture.	10
III	Peer to Peer Systems–Introduction, Napster and its legacy, Peer to Peer middleware, Routing overlays, Overlay case studies-Pastry, Tapestry, Application case studies-Squirrel, Ocean Store. Time and Global States-Introduction, Clocks, events and Process states, Synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging. Coordination and Agreement-Introduction, Distributed mutual exclusion, Elections, Multicast communication, consensus and related problems.	10
IV	Transactions and Concurrency Control-Introduction, Transactions, Nested Transactions, Locks, and Optimistic concurrency control, Timestamp ordering. Distributed Transactions-Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery.	10
V	Replication-Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data. Distributed shared memory, Design and Implementation issues, Consistency models.	10
VI	Distributed Multimedia Systems: Characteristics of Multimedia Data, Quality of Service Management, Resource management, Stream Adaptation. Case Study: BitTorrent and End System Multicast.	10
	Text Books: 1) Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education. 2. Distributed Systems, S.Ghosh, Chapman & Hall/CRC, Taylor & Francis Group, 2010.	
	Reference Books: 1. Distributed Systems – Principles and Paradigms, A.S. Tanenbaum and M.V. Steen, Pearson Education. 2. Distributed Computing, Principles, Algorithms and Systems, Ajay D. Kshemakalyani and Mukesh Singhal, Cambridge, rp 2010.	

Skill-III

Course Code	**3MCSW5**
Course Name	**Programming with .NET Technology**
Credits	**4**
Course Outcomes	**COs:** **Upon successful completion of the course, students would be able to** 1. Demonstrate the use of various Vb.net controls 2. Describe the procedural and object oriented paradigm with concepts of classes, functions, data and objects in VB.Net 3. Program to demonstrate the implementation of constructors, destructors and function overloading. 4. Use the syntax and semantics of ASP.Net. 5. Design web applications using ASP.Net

Sr. No.	Contents	Total Hrs
	The following information can be used as guidelines for basic understanding of .NET technology Example: 1. Write a Program in VB.Net to design and display Student marksheet 2. Write a Program in VB.Net to design and find Employee salary sheet (using list box, combo box controls) 3. Write a Program in VB.Net to design Electricity bill (using list box, combo box controls) 4. Design a simple Notepad application in VB.Net using MenuStrip, and RichTextBox that allows user to set font color and font style to text. Also display selected text on the textbox. 5. Design library system using ADO for database connectivity for ASP/VB.net 6. Design hotel management system using ADO. 7. Write a Program in ASP.Net using validation controls. 8. Write a Program in ASP.Net to demonstrate timer control, date picker controls.	

Semester-III

Elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
3MCSW6(1)	**Optimization Techniques**	**60**

Course Objectives (Cos)

Upon successful completion of the course, students would be able to

1. Formulate the linear and nonlinear programming problems.
2. Solve linear programming problems using Simplex method and its variants.
3. Construct and optimize various network models.
4. Construct and classify multi-objective linear programming problems.
5. Solve nonlinear programming problems

Unit	**Content**	**Hrs**
Unit I	Introduction to Optimization: Engineering application of Optimization – Statement of an Optimization problem – Optimal Problem formulation – Classification of Optimization problem. Optimum design concepts: Definition of Global and Local optima – Optimality criteria – Review of basic calculus concepts – Global optimality	10
Unit II	Optimization algorithms for solving unconstrained optimization problems – Gradient based method: Cauchy's steepest descent method, Newton's method, Conjugate gradient method. Optimization algorithms for solving constrained optimization problems – direct methods – penalty function methods – steepest descent method – Engineering applications of constrained and unconstrained algorithms.	10
Unit III	Modern methods of Optimization: Genetic Algorithms – Simulated Annealing – Ant colony optimization – Tabu search – Neural-Network based Optimization – Fuzzy optimization techniques – Applications. Use of Matlab to solve optimization problems.	10
Unit IV	Linear Programming: Geometry of linear programming, Graphical method, Linear programming (LP) in standard form, Solution of LP by simplex method, Exceptional cases in LP, Duality theory, Dual simplex method, Sensitivity analysis.	10
Unit V	Integer Programming: Branch and bound technique, Gomory's Cutting plane method. Network Models: Construction of networks, Network computations, Free Floats, Critical path method (CPM), optimal scheduling (crashing). Initial basic feasible solutions of balanced and unbalanced transportation problems, optimal solutions, assignment problem.	10
Unit VI	Nonlinear Programming: Unconstrained Optimization: unimodal functions, Fibonacci search method, Steepest Descent method. Constrained Optimization: Concept of convexity and concavity, Maxima and minima of functions of n- variables, Lagrange multipliers, Karush-Kuhn-Tucker conditions for constrained optimization	10
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity		
1. COs: This will discuss fundamental concepts and tools in discrete mathematics 2. Understand sets and perform operations and algebra on sets. Determine properties of relations, identify equivalence and partial order relations, sketch relations. Identify functions and determine their properties. 3. Cos: To assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour		
Text Books	1. Rao S. S. – 'Engineering Optimization, Theory and Practice' – New Age International Publishers – 2012 – 4th Edition 2.Chandra, S., Jayadeva, Mehra, A., Numerical Optimization and Applications, Narosa Publishing House, (2013). 3. Taha H.A., Operations Research-An Introduction, PHI (2007).	
Reference Books	1) Pant J. C., Introduction to optimization: Operations Research, Jain Brothers (2004) 2) BazaarraMokhtar S., Jarvis John J. and ShiraliHanif D., Linear Programming and Network flows, John Wiley and Sons (1990) 3) Swarup, K., Gupta, P. K., Mammohan, Operations Research, Sultan Chand & Sons, (2010). 4) H.S. Kasana and K.D. Kumar, Introductory Operations research, Springer publication, (2004)	

Prescribed Syllabus
Semester-III

Elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
3MCSW6(2)	**Cloud Computing**	**60**

Course Objectives (Cos)

Upon successful completion of the course, students would be able to

1: Describe the basic concept of Cloud Computing and Its Models.
2: Analyze and the application and virtualization infrastructures for cloud computing.
3: Exhibit in-depth understanding of key cloud-based services.
4: Understand the necessity of management activity at cloud environment.
5: Create Virtual environment using hypervisor.
6: Develop own cloud with desired functions.

Unit	**Content**
Unit I	**Fundamentals of Cloud Computing:** Definition and History, Cloud Characteristics, Cloud Advantages and Disadvantages, Cloud Provider, Cloud Consumer **Distributed Computing**: Client Server, Multitier Architecture **Parallel Computing**: Flynn's Taxonomy, SIMD vs MIMD, **Cloud-based RESTful API**: Principle, Components, working, Authentication Method, Benefits.
Unit II	**Cloud Architecture and Services:** NIST Cloud Computing Reference Architecture **Cloud Deployment Models:** Public Cloud, Private Cloud, Community Cloud, Hybrid Cloud. **Cloud Delivery Models:** Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS).
Unit III	**Cloud Enabling Technology**: Internet Service Provider (ISP) **Web Technology**: Basic Concept, Fundamental Elements (URL, HTTP, HTML,XML), Web Application Architecture. **Data Center**: Concept and Challenges, **Data Center Virtualization**: Basic Model of Virtualization, Benefits of Virtualization, Para Virtualization (O.S. Based), Full Virtualization (Hardware Based). **Virtualization Platform**: Xen and Virtual Box: Terms, Need, Structure, Merits and Demerits.
Unit IV	**Cloud Infrastructure and Management**: Virtual Server **Cloud Storage Device**: Cloud Storage Level, Cloud Storage Reference Model **Cloud Usage Monitor**: Monitoring Agent, Resource Agent, Polling Agent, Resource Replication **Cloud Management**: Need, Cloud Management Tasks, Features, Resource Provisioning, Resource Management, SLA Management.
Unit V	**Cloud Building Platform**: **Eucalyptus**: Architecture Design, Components (Control Plane): Cloud Controller, Cluster Controller, Storage Controller, Node Controller, Client Interface, Features, Advantages, Installation Procedure **OpenStack**: Architecture, Components, OpenStack Landscape, Installation and manage instances process, Features, Advantages.
Unit VI	**Cloud Security Essentials**: **Basic Terms**: Confidentiality, Integrity, Authenticity, Availability, Threats, Vulnerabilities, and Risks **Threat Agents**: Anonymous Attacker, Malicious Service Agent, Trusted Attacker, Malicious Insider **Cloud Security Threats**: Traffic Eavesdropping, Malicious Intermediary, Denial of Service, Insufficient Authorization, Virtualization Attack **Encryption**: Symmetric Encryption, Asymmetric Encryption, Hashing, Digital Signature, Public Key Infrastructure (PKI), **Identify and Access Management** (IAM): Authentication and Authorization with IAM.
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	
**Activities	1.

Course Material/Learning Resources

Text books:

1. "Cloud Computing: Methodology, System, and Application", Lizhe Wang, CRC Press, 2017.
2. "Cloud Computing a Practical Approach", Toby Velte et.al., McGraw Hill, 2017.
3. "Cloud Computing and Virtualization", Dac-Nhuong Le et.al., Wiley, 2018.
4. "Design and Use of Virtualization Technology in Cloud Computing",P. Kumar Das, IGI Global, 2018.

Reference Books:

1. "Cloud Computing: Concept, Technology and Architecture", Thomas Erl et.al., Pearson, 2013.
2. "Cloud Computing for Science and Engineering", Ian Foster and Dennis, MIT Press, 2017.
3. "Cloud Computing: From Beginning to End", Ray Rafaels, 2018.
4. "Virtualization Essentials", Matthew Portnoy, Sybex, 2012.

5. "The Value Of Virtualization And Cloud Computing", Manny Vergara, 2013.

Weblink to Equivalent MOOC on SWAYAM if relevant:

1. https://onlinecourses.nptel.ac.in/noc23_cs89/preview
2. https://onlinecourses.nptel.ac.in/noc23_cs90/preview

Prescribed Syllabus

Semester-III

Elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
3MCSW6(3)	**Software Project Management**	**60**

Course Objectives (Cos)

This Course is intended to provide the students with an overall view over Software Engineering Discipline and with insight into the processes of software development.

Upon successful completion of the course, students would be able to

1. Understand the various methods of Cost Estimation.
2. Study Software Quality Management.
3. Study Software Metrics.

Unit	**Content**
Unit I	**Project Concepts and Its Management** Project life cycle models-ISO 9001 model - Capability Maturity Model - Project, Planning-Project tracking-Project closure - Evolution of Software Economics – Software Management Process Framework: Phases, Artifacts, Workflows, Checkpoints – Software Management Disciplines: Planning / Project Organization and Responsibilities / Automation / Project Control – Modern Project.
Unit II	**Project Life Cycle and Effort Estimation:** Software process and Process Models – Choice of Process models - Rapid Application development – Agile methods – Dynamic System Development Method – Extreme Programming– Managing interactive processes – Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points - COCOMO II - a Parametric Productivity Model.
Unit III	**Activity Planning and Risk Management:** Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Formulating Network Model – Forward Pass & Backward Pass techniques – Critical path (CRM) method – Risk identification – Assessment – Risk Planning – Risk Management – – PERT technique – Monte Carlo simulation – Resource Allocation – Creation of critical paths – Cost schedules.
Unit IV	**Project Management and Control** Framework for Management and control – Collection of data – Visualizing progress – Cost monitoring – Earned Value Analysis – Prioritizing Monitoring – Project tracking – Change control – Software Configuration Management – Managing contracts – Contract Management.
Unit V	**Staffing In Software Projects:** Managing people – Organizational behavior – Best methods of staff selection – Motivation – The Oldham – Hackman job characteristic model – Stress – Health and Safety – Ethical and Professional concerns – Working in teams – Decision making – Organizational structures – Dispersed and Virtual teams – Communications genres – Communication plans – Leadership.
Unit VI	**Software and System Quality Management**: Overview of ISO 9001, SEI Capability Maturity Model, McCalls Quality Model, Six Sigma, Formal Technical Reviews, Tools and Techniques for Quality Control, Pareto Analysis, Statistical Sampling, Quality Control Charts and the seven Run Rule. Modern Quality Management, Juran and the importance of Top management, Commitment to Quality, Crosby and Striving for Zero defects, Ishikawa and the Fishbone Diagram.
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	
4. COs: This will discuss fundamental concepts and tools in discrete mathematics 5. Understand sets and perform operations and algebra on sets. Determine properties of relations, identify equivalence and partial order relations, sketch relations. Identify functions and determine their properties. 6. Cos: To assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour	

Text Books	1. Ramesh Gopalaswamy, Managing and Global Software Projects, Tata McGraw Hill, 2017. 2. Neal Whitten, Managing Software Development Projects, John Wiley & Sons, Inc., 2nd Ed., 1995.
Reference Books	1. Demarco, T. and Lister, T. Peopleware: Productive Projects and Teams, 2nd Ed., Dorset House, 1999. 2. Royce, W. Software Project Management: A Unified Framework, Addison-Wesley, 1998. Demarco, T. and Lister, T. Peopleware: Productive Projects and Teams, 2ndEd., Dorset House,1999. 3. Fenton, N.E., and Pfleeger, S.L. Software Metrics: A Rigorous and Practical Approach, Revised Brooks Cole, 1998. 4. Kaplan, R.S., Norton, D.P. The Balanced Scorecard: Translating Strategy into Action, Harvard Busine ss School Press, 1996. 5. Boehm, B. W. Software Risk Management: Principles and Practices in IEEE Software, January 1991, pp32-41. 6. Grant, J.L. Foundations of Economic Value Added, John Wiley & Sons, 1997.

Prescribed Syllabus

Semester-III

Elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
3MCSW6(4)	**Bioinformatics Techniques**	**60**

Course Objectives (Cos)

Upon successful completion of the course, students would be able to

1. gain knowledge about various Biological databases that provide information about nucleic acids and protein.
2. Use Biological databases and database systems.
3. Compare and use types and Biological data and database search tools.
4. Describe about the different types of Biological databases.
5. Explain about different types of protein and other organism specific databases.
6. understand pathway and enzyme databases, Sequence submission tools.

Unit	**Content**
Unit I (Biological Databases	Primary Sequence & Structure Databases: Genbank, SwissProt/Uniprot, EMBL, PIR, PDB, MMDB, NDB, CSD, KEGG etc. Derived (Secondary) Databases of Sequences and Structure: Prosite, PRODOM, PRINTS, Pfam, BLOCK, INTERPRO etc. SSOP, CATH, DSSP, FSSP, RNAbase, Genome Databases (at NCBI, EBI, TIGR, SANGER), High- throughput genomics sequences (EST, STS, GSS), ENSEMBL.
Unit II (Advanced techniques)	Algorithms for searching sequence patterns: MeMe, PHI-BLAST, SCanProsite and PRATT; Algorithms for generation of sequence profiles: Profile Analysis method of Gribskov, HMMER, PSI-BLAST; Basic concepts on identification of disease genes, role of bioinformatics-OMIM database, reference genome sequence, integrated genomic maps, gene expression profiling; identification of SNPs, SNP database (DbSNP). Role of SNP in Pharmacogenomics, SNP arrays.
Unit III (Signals in DNA)	Introduction, DNA linguistics, Convey equation, Consensus, CG-islands, HMM, Gibbs sampling, Gene Promoter, Enhancers, Gene Prediction – introduction, statistical approaches, Spliced alignment, Reverse gene finding, some other problems.
Unit IV (Genome Rearrangement)	Introduction, Deletion, Insertion, Inversion, Translocation, Capping chromosomes, Caps and tails, Genome duplication, Genome rearrangement tools, Synteny and Rearrangement.
Unit V (DNA microarray)	DNA microarray: database and basic tools, Gene Expression Omnibus (GEO), Array Express, SAGE databases DNA microarray: understanding of microarray data, normalizing microarray data
Unit VI (Gene detection techniques)	Detecting differential gene expression, correlation of gene expression data to biological process and computational analysis tools (especially clustering approaches)
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	

7. COs: This will discuss fundamental concepts and tools in discrete mathematics 8. Understand sets and perform operations and algebra on sets. Determine properties of relations, identify equivalence and partial order relations, sketch relations. Identify functions and determine their properties. 9. Cos: To assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour	
	Reference Books: 1. Baxevanis, A.D. and Francis Ouellellette, B.F. (1998) "Bioinformatics– a practical guide to the analysis of genes and proteins" John Wiley and Sons 2. Des Higgins, Willie R. Taylor, Willie Taylor (2000) "Bioinformatics: sequence, structure, and databanks : a practical approach" Oxford University Press 3. Mount, D. (2004) "Bioinformatics: Sequence and Genome Analysis"; Cold Spring Harbor Laboratory Press, New York. (ISBN 0-87969- 712-1) 4. Sharma, V. Munjal, A. and Shankar, A. (2008) "A text book of Bioinformatics" first edition, Rastogi Publication, Meerut – India. 5. Stanley Letovsky (1999) "Bioinformatics: databases and systems" Springer

Prescribed Syllabus

Semester-III
Elective
Subjects

Elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
3MCSW6(5)	**Information System Security**	**60**

COs:

Upon completion of this course successfully, students would be able to

1. discuss fundamental concepts and tools in discrete mathematics
2. Understand sets and perform operations and algebra on sets.
3. Determine properties of relations
4. identify equivalence and partial order relations, sketch relations
5. Identify functions and determine their properties.

Unit	**Content**
Unit I	Security and Risk Management (Security, Risk, Compliance, Law, Regulations, and Business Continuity): Confidentiality, Integrity, and availability concepts - Security governance principles – Compliance - Legal and regulatory issues -Professional ethic - Security policies, standards, procedures and guidelines
Unit II	Asset Security (Protecting Security of Assets): Information and asset classification -Ownership (e.g. data owners, system owners) - Protect privacy - Appropriate retention - Data security controls - Handling requirements (e.g. markings, labels, storage)
Unit III	Security Engineering (Engineering and Management of Security) - Engineering processes using secure design principles - Security models fundamental concepts - Security evaluation models - Security capabilities of information systems -Security architectures, designs, and solution elements vulnerabilities – Web based systems vulnerabilities - Mobile systems vulnerabilities - Embedded devices and cyber physical systems vulnerabilities – Cryptography - Site and facility design secure principles – Physical Security
Unit IV	Communication and Network Security (Designing and Protecting Network Security): Secure network architecture design (e.g. IP & non - IP protocols, segmentation) - Secure network components - Secure communication channels - Network attacks
Unit V	Application Security & Vulnerabilities Application Security – Application Vulnerabilities - OWASP Top 10 Web Security Vulnerabilities - Unvalidated input, Broken access control, Broken account/session management, Cross-site scripting (XSS) flaws, Buffer overflows - SQL Injection flaws, Improper error handling, Insecure storage, Denial -of service, Insecure configuration management –Insecure File Handling

Unit VI	Identity and Access Management (Controlling Access and Managing Identity) - Physical and logical assets control -Identification and authentication of people and devices - Identity as a service (e.g. cloud identity) – Third-party identity services (e.g. on- premise) - Access control attacks - Identity and access provisioning lifecycle (e.g. provisioning review)
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	

10. Cos: To assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour	
	Reference Book: 1. James M. Stewart, Ed Tittel, Mike Chapple 'CISSP: Certified Information Systems Security Professional Study Guide', Wiley 2008. 2. Web application security scanners, http: // www. Window security. com / software/Web-Application-Security. 3. The Open Web Application Security Project, http://www.owasp.org

Course Material/Learning Resources

Weblink to Equivalent MOOC on SWAYAM if relevant:

Weblink to Equivalent Virtual Lab if relevant:

Any pertinent media (recorded lectures, YouTube, etc.) if relevant:

Prescribed Syllabus

Semester-III

Core Subjects

Laboratories

Course Code	**3MCSW7**
Course Name	**Lab-V 1,2 – Web development tools / CMS Tools**
Total Credits	**2**
Course Outcomes	Course Outcome: Upon successful completion of the course, students would be able to • design Web Applications • develop a Web Applications • work with Joomla • work with Wordpress • work with Drupal

	Contents	**Total Hours**

	The sample list of programs is given below. This list can be used as a guideline for problem statements but the scope of the laboratory should not be limited to the same. Aim of the list is to inform about minimum expected outcomes. (Min 15 programs) 1. WAP in WordPress to perform steps of Installation. 2. WAP in WordPress to install and activate theme. 3. WAP in WordPress to perform steps of Customization. 4. WAP in WordPress to Create a Page. 5. WAP in WordPress to create a Blog. 6. WAP in WordPress to create a Slider .(Using Smart Slider Plugin) 7. WAP in WordPress to Backup and restore file. 8. WAP in Joomla process of installation. 9. WAP in Joomla to change Templates Styles (site). 10. WAP in Joomla to Create the Article. 11. WAP in Joomla to Create Menus. 12. WAP in Joomla to create Custom Modules on Website. 13. WAP in Joomla to create Mass E-mailing. 14. WAP in Drupal to Perform steps of Installation. 15. Create Content, Modify Content, Publish Content, Menu Management. 16. WAP in Drupal to Create Taxonomies.	

Prescribed Syllabus
Semester-III

Course Code	**3MCSW8**
Course Name	**Lab-VI 3-Tools for ML / SciLab / Python**
Total Credits	**2**
Course Outcomes	**Course Outcomes:** **Upon successful completion of the course, students would be able to** **1.** Make use of Data sets in implementing the machine learning algorithms **2.** Implement the machine learning concepts and algorithms in any suitable language of choice.

	Contents	**Total Hours**
	The sample list of programs is given below. This list can be used as a guideline for problem statements but the scope of the laboratory should not be limited to the same. Aim of the list is to inform about minimum expected outcomes. (Min 15) 1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file. 2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples. 3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. 4. Use an appropriate data set for building the decision tree and apply this knowledge 5. To classify a new sample. 6. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets. 7. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets. 8. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem. 9. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs	

Prescribed Syllabus
Semester-IV Core Subjects

Course Code	**4MCSW1**
Course Name	**Software Testing**
Total Credits	**4**
Course Outcomes	After completion of this course successfully, students would be able to 1. Understand fundamental concepts in software testing 2. Distinguish characteristics of various testing methods. 3. Apply software testing knowledge and engineering methods. 4. Understand the process of applying tests to software and the fundamental components of a test case. 5. Derive test cases from software requirement specifications - including being able to partition input and output domains, form test specifications, and identify valid combinations of input. 6. Understand the role and process of software localization in software testing 7. Understand the importance of test planning and management in success of a project

Units	Contents	Hrs
I	Principles of Testing, Software development models, Agile software development and testing, Software Testing Life Cycle, Quality Assurance and Quality Control, Verification and Validation, Context driven testing, Software testing best practices	10
II	Types of testing: Black Box and White Box, Functional and Non-functional, White Box: Static Testing, Structural Testing, Challenges, Black Box: Positive-Negative Testing, Boundary Value Analysis, Equivalence Partitioning, Decision Tables, Challenges	10
III	Integration Testing: Integration Testing Types - Top-down integration, Bottom-up integration, Bi-directional integration, System integration, Choosing suitable integration method Regression Testing: Circumstances for regression testing, Types o Regression Testing – Smoke test vs. Sanity test, Test case classification, Test case selection, Steps in regression testing Performance Testing: Factors governing performance testing, Types of performance testing, Methodology, Identifying test tools and environment, Capacity planning, Defining performance criteria, Performance Tuning, Challenges	10
IV	System Testing: Types of System Testing, Functional System Testing, Non-functional System Testing, Focus and Criteria for System testing, Methodology/Process of System Testing Usability Testing: Approach to usability, Comparative usability testing, Exploratory usability testing, Usability Testing methods, Quality factors for usability, Tools and Lab setup for Usability Testing, Methodology/Process of usability testing Accessibility Testing: Usability vs. Accessibility, Types of Accessibility, Accessibility and the law, Challenges in Accessibility testing,	10
V	Internationalization: Difference between Globalization, Internationalization and Localization, Language and Locale, Deciding scope of testing, Internationalization validation, Language Testing, Localization Testing, Using Tools for internationalization Automation: Skills needed for automation, Scope of automation, Criteria for choosing automation areas, Design considerations, Process of Automation, Automation Tools and criteria for choosing a tool, Challenges in automation	10
VI	Test Planning: Preparing a Test plan, Deciding Test Strategy, Setting up criteria, Roles and Responsibilities, Test deliverables, Scheduling Test Management: Test infrastructure management, Test resource management, Test management process, Phases – Planning and Execution, Integrating with product release, Test report and evaluation, Benefits of Test management Metrics and Measurement: Understanding Measurement and Metrics, Metrics vs. Measurement, Need for measurement of metrics, Types of Metrics – Project\Product\Process Metrics, Metrics Life Cycle, Base and Calculated Metrics, Important test matrices and formulas	10

	Text Books: 1) Software Testing Principles and Practices by Srinivasan Desikan, Gopalaswamy Ramesh, PEARSON	
	Reference Books: 1. Software Testing by Ron Patton, SAMS 2. Software Testing: A Craftsman's Approach by Paul C. Jorgensen, CRC Press 3. Agile Testing: A Practical Guide for Testers and Agile Teams 4. by Lisa Crispin and Janet Gregory, Addison-Wesley 5. Software Test Automation by Mark Fewster and Dorothy Graham, Addison-Wesley	

Code of the Course/Subject	Title of the Course/Subject	(Total Number of Periods)
4MCSW2	**Big Data Analytics**	**60**

Course Outcomes: On completion of this course, students would be able to:

CO1: Get acquainted with the basics of Big Data Technology.
CO2: Get the basics understanding of Data Analytics**.**
CO3: Work with Hadoop Distributed File System to handle the Big Data.
CO4: Use the tool MapReduce to analyze the data efficiently.
CO5: Use the open source framework Hive to process the large dataset.
CO6: Understand the use of MongoDB a NoSQL database management system.

Unit	Contents	Hrs
Unit I	**Fundamentals of Big Data:** Concepts and Terminologies: *Data, Big Data, Data sets, Data Analysis and Data Analytics*, Evolution of Big Data, Big Data Characteristics, Types of Data, Big Data Storage Concepts: *Clusters, File Systems & Distributed File Systems, NoSQL, Sharding and Replication*, ACID, CAP Theorem, BASE.	10
Unit II	**Basics of Data Analytics:** What is Data Analytics?, Types of Data Analytics, Business Intelligence and Data Mining Cycle, Pattern Recognition: *Types of Pattern, Finding a Pattern, Uses of Pattern*. Data Processing Chain, Business Intelligence Vs. Big Data, Big Data Analytics Data Life Cycle, Big Data Analysis Techniques: *Quantitative Analysis, Qualitative Analysis, Data Mining, Statistical Analysis, Machine Learning, Semantic Analysis, Visual Analysis*.	10
Unit III	**Introduction to Hadoop and HDFS:** History and overview of Hadoop, Data Storage and Data Analysis, Comparison with RDBMS, Greed Computing and Volunteer Computing, Apache Hadoop and Hadoop Ecosystem, Hadoop Releases and Compatibility, Installing and Configuring Hadoop on Windows. **HDFS**: The Design of HDFS, HDFS Concepts: *Blocks, Namenodes and Datanodes, HDFS Federation*, Command Line Interface: *Basic File System Operations,* Interfaces: *HTTP, C and FUSE*	10
Unit IV	**MapReduce**: What is MapReduce? Why MapReduce? And How MapReduce works? Installing MapReduce, MapReduce Algorithm, MapReduce Partitioner, MapReduce Combiners, Map Reduce Types and Formats, Anatomy of MapReduce Job run, Shuffle and Sort: The Map Side and The Reduce Side. MapReduce Features: Counters, Sorting, Joins	10
Unit V	**Apache Pig**: Introduction, Need of Apache Pig, Features of Apache Pig, Apache Pig Vs. MapReduce, Apache Vs. SQL, Installing Apache Pig, **Pig Latin:** *Structures*, *Statements, Expressions, Data Types, Functions*. **Data Processing Operators**: *Loading and Storing Data, Grouping and Joining the Data, Sorting Data.*	10
Unit VI	**Apache Hive and Introduction to MongoDB**: Architecture and Features of Apache Hive, Installing Apache Hive, Hive-Data Types, Creating Database, Dropping Database, Creating Table, Altering Table, Dropping Table, Hive-Partitions, HiveQL. MongoDB Structure, How MongoDB is different from RDBMS, Terminologies: Documents, Collection and Database, MongoDB Shell, Data Types, Creating, Updating and Deleting Documents.	10

Text books: format (Title, Author, Publisher, Edition)

1. Big Data Fundamentals: Concepts, Drivers & Techniques by Thomas Erl, Wajid Khattak & Paul Buhler Publication: Prentice Hall, ISBN-13: 978-0-13-429107-9 ISBN-10: 0-13-429107-7
2. Big Data Principles and Paradigms by Rajkumar Buyya, Rodrigo Publication: Elsevier Inc.
3. Hadoop: The Definitive Guide by Tom White Publication: O'Really 3rd Edition

Reference Books: format (Title, Author, Publisher, Edition)

1. A General Introduction to Data Analytics by Moreira, Carvalho, Horvath Publication: Wiley & Sons Inc.
2. Data Analytics by Dr. Anil Maheshwari Publication: McGraw Hill Education
3. MongoDB: The Definitive Guid by Kristina Chodorow and Michael Dirolf Publication: O'Rielly

Links of MOOC[SWAYAM]:

https://onlinecourses.nptel.ac.in/noc20_cs92/preview
https://onlinecourses.swayam2.ac.in/arp20_ap10/preview
https://www.coursera.org/learn/introduction-to-big-data-with-spark-hadoop

Syllabus Prescribed for First/Second -Year
PG Programme: M.Sc. (Computer Software) **Semester: I / II / III/ IV**

Code of the Course/Subject	Title of the Course/Subject	(Total Number of Periods)
4MCSW3 -	**Internet of Things (IOT) Technology**	60

Course Outcomes: On completion of this course, students would be able to:
CO1: Understand the various concepts, terminologies and architecture of IoT systems.
CO2: Understand and apply various protocols for design of IoT systems
CO3: Use sensors and actuators for design of IoT.
CO4: Use various techniques of data storage and analytics in IoT.
CO5: Understand and develop programming with Arduino.
CO6: Understand and develop programming with Raspberry Pi.

Unit	Contents	Periods
Unit I	**Internet of Things**: An Overview, IoT Conceptual Framework, IoT Architecture View, Technology behind IoT, Sources of IoT, M2M Communication., IoT/M2M system layers and design standardization. **Communication Technologies:** Wireless Communication Bluetooth, ZigBee, WiFi, RF Links, Mobile Internet, Wired Communication, serial communication, SPI Layered Protocol, Ethernet.	**10 Periods**
Unit II	**IoT Protocols**: MQTT, CoAP, XMPP, OSGi Architecture/Services. Internet based communication, Internet protocol, IP addressing in the IoT, Media Access Control, TCP/IP, Overview of each Layer, HTTP. **Data Acquiring, Organizing, Processing and Analytics**: Data acquiring and storage, Organizing the data, transaction, business process, integration and enterprise system, data analytics.	**10 Periods**
Unit III	**Sensors Networks**: Definition, Types of Sensors, Types of Actuators, Examples and Working. **Wireless Sensor Networks**: History and Context, the node, Connecting nodes, Networking Nodes, WSN and IoT. **IoT Development Boards**: Arduino IDE and Board Types	**10 Periods**
Unit IV	**Arduino:** The Development Environment, Writing Arduino Software, The Arduino Sketch., Introduction to programming for Arduino: Variables, functions and statements, array, strings, Input and Output. Standard arduino library, Interfacing LED, push button and buzzer with Arduino, Interfacing Arduino with LCD.	**10 Periods**
Unit V	**Raspberry Pi**: Interfacing Hardware with the Raspberry Pi, Raspberry Pi Remote Access, operate the Raspberry Pi in "headless mode", Bash Command line, operating Raspberry Pi without needing a GUI interface. Python on Raspberry Pi, Python Programming Environment, Python Expressions, Strings, Functions and Function arguments, Lists, List Methods, Control Flow. Communication with devices through the pins of the Raspberry Pi, RPi.GPIO library, Python Functions, setting up the pins, General purpose IO Pins, Protocol Pins, GPIO Access	**10 Periods**
Unit VI	**Applications of IoT:** Home Automation, Smart Cities, Energy, Retail Management, Logistics, Agriculture, Health and Lifestyle, Industrial IoT, Legal challenges, IoT design Ethics, IoT in Environmental Protection.	**10 Periods**
Text books: format (Title ,Author, Publisher, Edition) 1. Raj Kamal, "INTERNET OF THINGS Architecture and Design Principles" McGraw Hill Education (India) Private Limited. 2. Simon Monk, "Programming the Raspberry Pi: Getting Started with Python", January 2012, McGraw Hill Professional. 3. Simon Monk, "Programming Arduino: Getting Started with Sketches", McGraw Hill Professional.		
Reference Books: format (Title ,Author, Publisher, Edition) 1. Waher, Peter. Learning internet of things. Packt Publishing Ltd, 2015. 2. Hakima Chaouchi, — "The Internet of Things Connecting Objects to the Web" ISBN : 978-1- 84821-140-7, Wiley Publications 3. Olivier Hersent, David Boswarthick, and Omar Elloumi, — "The Internet of Things: Key Applications and Protocols", WileyPublications 3. Vijay Madisetti and ArshdeepBahga, — "Internet of Things (A Hands-on-Approach)", 1 st Edition, VPT, 2014. 4. J. Biron and J. Follett, "Foundational Elements of an IoT Solution", O'Reilly Media, 2016. 5. Keysight Technologies, "The Internet of Things: Enabling Technologies and Solutions for Design and Test", Application Note, 2016.		

Swayam Web Links:
1. https://onlinecourses.nptel.ac.in/noc19_cs65/preview
2. https://onlinecourses.nptel.ac.in/noc21_ee85/preview
3. https://onlinecourses.swayam2.ac.in/arp19_ap52/preview

Video Links:
1. **https://www.youtube.com/watch?v=JWQjtV4I9y0**

Prescribed Syllabus
Semester-IV Core Subjects

Course Code	**4MCSW4**
Course Name	**Cyber Security and Digital Forensics**
Total Credits	**4**
Course Outcomes	**Course Outcome:** After completion of this course successfully, students would be able to **1.** Use cryptographic techniques in secure application development. **2.** Apply methods for authentication, access control, intrusion detection and prevention. **3.** apply the scientific method for security assessment **4.** develop computer forensics awareness.

Units	Contents	Hrs
I	Security Fundamentals: An Overview of Information Security: The Basic Components, Threats, Policy and Mechanism, Assumptions and Trust, Assurance, Operational Issues, Human Issues, Security nomenclature. Access Control Matrix, Security Policies: Confidentiality, Integrity, Availability Policies and Hybrid Policies, OS Security	10
II	Issues in Security Management and Cyber Laws : Overview, Risk identification, Risk Assessment, Risk Control Strategies, Quantitative vs. Qualitative Risk Control Practices. Risk Management. Laws and Ethics in Information Security, Codes of Ethics, Protecting programs and data Cybercrime and Information security, Classification of Cybercrimes, The legal perspectives- Indian perspective, Global perspective, Categories of Cybercrime, Types of Attacks, a Social Engineering, Cyber stalking, Cloud Computing and Cybercrime.	10
III	Key Management and Secure Communication: Public Key Infrastructure(PKI), X.509 Certificate, Needham Schroeder algorithm and Kerberos. IP Security: IPv6 and IPSec, Web Security: SSL, HTTPS, Mail Security: PGP, S/MIME . Firewall : Different Types and Functionalities.	10
IV	Cyber Attacks & Countermeasures: Phishing, Password Cracking, Key-loggers and Spywares, Types of Virus, Worms, DoS and DDoS, SQL injection, Buffer Overflow, Spyware, Adware and Ransomware. Antivirus and other security measures Intrusion Detection System : IDS fundamentals, Different types of IDS. Intrusion Prevention.	10
V	Introduction to Digital Forensics : Digital forensics: Introduction – Evidential potential of digital devices: closed vs. open systems, evaluating digital evidence potential- Device handling: seizure issues, device identification, networked devices and contamination.	10
VI	Analysis of Digital Forensic Techniques: Digital forensics examination principles: Previewing, imaging, continuity, hashing and evidence locations- Seven element security model- developmental model of digital systems- audit and logs- Evidence interpretation: Data content and context.	10
	Text Books: 1) William Stallings, Computer Security: Principles and Practices, Pearson 6 Ed, ISBN 978-0-13-335469-0 2) Nina Godbole, Sunit Belapure , Cyber Security- Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiely India Pvt.Ltd, ISBN- 978-81-265-2179-1	
	Reference Books: 1. Iosif I. Androulidakis, Mobile phone security and forensics: A practical approach, Springer publications, 2012. 2. Andrew Hoog, Android Forensics: Investigation, Analysis and Mobile Security for Google Android, Elsevier publications, 2011. 3. Angus M. Marshall, Digital forensics: Digital evidence in criminal investigation, John – Wiley and Sons, 2008 4. Gregory Kipper, Wireless Crime and Forensic Investigation, Auerbach Publications.	

Skill

CourseCode	**4MCSW5**	
CourseName	**Android Programming**	
Credits	**4**	
Course Outcomes	**Upon successful completion of the course, students would be able to** 1. Understand Android OS 2. Understand Android App structure and its components 3. Learn to use Android Emulator for development and testing 4. Develop basic Android App using Android Studio. 5. Use database to store data from an Android App	
Sr. No.	Contents	Total Hrs
	The following information can be used as guidelines for basic understanding of Android programming	
	Android Introduction: What is an Android App? How to set up the development Environment What is the .apk file extension and Dalvik Virtual Machine. Application Structure Important topics covered in this section are: AndroidManifest.xml Uses-SDK R.java and Resources Layouts and & Drawable Resources Activities and Active Life Cycle Emulator- Android Virtual Device Setup and Launch emulator Create AVDs Running app on Emulator Basic UI Design Form widgets Text Fields Layouts [dip, dp, sip, sp] versus px Storing Data Shared Preferences Using SQLite database Menu Option menu Context menu Sub menu Menu from XML Menu via code Intents Explicit Intents Implicit intents	

	Lab Assignments: 1. Develop an Android app- Hello Android 2. Develop an Android app- Greetings 3. Develop an Android app to use Toasts & SnackBar 4. Develop an Android app to generate and display random numbers 5. Develop an Android app to demonstrate passing data across activities using Intents 6. Develop an Android app to demonstrate Data Persistence using Shared Preferences 7. Develop an Android app to demonstrate use of Pictures and Menus 8. Develop an Android app to play a song in Media Player 9. Develop an Android app to play a Video using VideoView 10. Develop an Android app to demonstrate Basic Calculator App 11. Develop an Android app to demonstrate use of Graphics Develop an Android app to create Basic Login form	

elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
4MCSW6(1)	**Parallel Computing**	**60**

COs:
After Completion of this Course, students would be able to
1. **Understand Parallel Processing Architectures**
2. **Analyse Data Dependency**
3. **Analyse Process model under UNIX Algorithms for Parallel Machines**
4. **Understand Message Passing Programming**
5. **Debug parallel programmes**
6. **Describe Other Parallelism Paradigms**

unit	**Content**
Unit I	Introduction: Parallel Processing Architectures: Parallelism in sequential machines, Abstract model of parallel computer, Multiprocessor architecture, Pipelining, Array processors. Programmability Issues: An overview, Operating system support, Types of operating systems, Parallel programming models, Software tools
Unit II	Data Dependency Analysis: Types of dependencies loop and array dependences, Loop dependence analysis, Solving diophantine equations, Program transformations Shared Memory Programming: General model of shared memory programming,
Unit III	Process model under UNIX Algorithms for Parallel Machines: Speedup, Complexity and cost, Histogram computation, Parallel reduction, Quadrature problem, Matrix multiplication, Parallel sorting algorithms, Solving linear systems, Probabilistic algorithms
Unit IV	Message Passing Programming: Introduction, Model, Interface, Circuit satisfiability, Introducing collective, Benchmarking parallel performance Parallel Programming languages: Fortran90, nCUBE C, Occam, C-Linda
Unit V	Debugging Parallel Programs: Debugging techniques, Debugging message passing parallel programs, Debugging shared memory parallel programs. Memory and I/O Subsystems: Hierarchical memory structure, Virtual memory system, Memory allocation and management, Cache allocation and management, Cache memories and management, Input output subsystems
Unit VI	Other Parallelism Paradigms: Data flow computing, Systolic architectures, Functional and logic paradigms, Distributed shared memory Performance of Parallel Processors: Speedup and efficiency, Amdahl's law, Gustafson- Barsis's law, Karf-Flatt metric, Isoefficiency metric

***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity

COs: This will discuss fundamental concepts and tools in discrete mathematics

13. Understand sets and perform operations and algebra on sets. Determine properties of relations, identify equivalence and partial order relations, sketch relations. Identify functions and determine their properties.
14. Cos: To assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour

Course Material/Learning Resources

Text books:

1. Hawang Kai and Briggs F. A., "Computer Architecture and Parallel Processing", McGraw Hill
2. Jorden H. F. and Alaghaband G., "Fundamentals of Parallel Processing"
3. M.J. Quinn, "Parallel Programming", TMH

Reference Books:

1. Shasikumar M., "Introduction to Parallel Processing", PHI
2. Wilson G.V., "Practical Parallel Programming", PHI
3. D. E. Culler, J.P. Singh, A. Gupta, "Parallel Computer Architecture", Morgan KaufmanWeblink to Equivalent MOOC on SWAYAM if relevant:

Elective-I **Code of the Course/Subject**	**Title of the Course/Subject**	**(Total Number of Hours)**
4MCSW6(2)	**Image Processing**	**60**

COs:
After Completion of this Course, students would be able to

1. **Understand Digital Image Processing Systems**
2. **Implement 2 –D Image Transforms**
3. **Analyse Image Enhancement in the Spatial Domain**
4. **Understand Wavelets and Multiresolution Processing**
5. **Analyse Morphological Image Processing**
6. **Implement Image Data Compression**

Unit	**Content**
Unit I	Digital Image Processing Systems: Introduction, Structure of human eye, Image formation in the human eye, Brightness adaptation and discrimination, Image sensing and acquisition, Storage, Processing, Communication, Display. Image sampling and quantization, Basic relationships between pixels
Unit II	Image Transforms (Implementation): Introduction to Fourier transform, DFT and 2-D DFT, Properties of 2-D DFT, FFT, IFFT, Walsh transform, Hadamard transform, Discrete cosine transform, Slant transform, Optimum transform: Karhunen - Loeve (Hotelling) transform.
Unit III	Image Enhancement in the Spatial Domain: Gray level transformations, Histogram processing, Arithmetic and logic operations, Spatial filtering: Introduction, Smoothing and sharpening filters Image Enhancement in the Frequency Domain: Frequency domain filters: Smoothing and Sharpening filters, Homomorphic filtering
Unit IV	Wavelets and Multiresolution Processing: Image pyramids, Subband coding, Haar transform, Series expansion, Scaling functions, Wavelet functions, Discrete wavelet transforms in one dimensions, Fast wavelet transform, Wavelet transforms in two dimensions.
Unit V	Morphological Image Processing: Introduction, Dilation, Erosion, Opening, Closing, Hit-or-Miss transformation, Morphological algorithm operations on binary images, Morphological algorithm operations on gray-scale images
Unit VI	Image Data Compression: Fundamentals, Redundancies: Coding, Interpixel, Psycho-visual, Fidelity criteria, Image compression models, Error free compression, Lossy compression, Image compression standards: Binary image and Continuous tone still image compression standards, Video compression standards.
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	

Course Material/Learning Resources

Text book & Reference Books :

1. R.C.Gonsales R.E.Woods, “Digital Image Processing”, Second Edition, Pearson Education
2. Anil K.Jain, “Fundamentals of Image Processing”, PHI.
3. William Pratt, “Digital Image Processing”, John Wiley
4. Milan Sonka,Vaclav Hlavac, Roger Boyle, “Image Processing, Analysis, and Machine Vision” Thomson Learning.
5. N Ahmed & K.R. Rao, “Orthogonal Transforms for Digital Signal Processing” Springer .
6. B. Chanda, D. Dutta Majumder, “Digital Image Processing and Analysis”, PHI. Reference Books:

lective-I Code of the Course/Subject	Title of the Course/Subject	(Total Number of Hours)
4MCSW6(3)	**Block-chain Technology**	**60**

Course Objectives (Cos)

Upon completion of this course, the students will be able to

1. discuss and describe the history, types and applications of Blockchain
2. Explain cryptography and Consensus algorithms.
3. Create and deploy projects using Web3j.
4. Implement an ICO on Ethereum
5. Design blockchain based application with Swarm and IPFS

Unit	Content
Unit I	INTRODUCTION TO BLOCKCHAIN : Distributed DBMS – Limitations of Distributed DBMS, Introduction to Block chain – History, Definition, Distributed Ledger, Blockchain Categories – Public, Private, Consortium, Blockchain Network and Nodes, Peer-to-Peer Network, Mining Mechanism, Generic elements of Blockchain, Features of Blockchain, and Types of Blockchain.
Unit II	BLOCKCHAIN ARCHITECTURE : Operation of Bitcoin Blockchain, Blockchain Architecture – Block, Hash, Distributer P2P, Structure of Blockchain- Consensus mechanism: Proof of Work (PoW), Proof of Stake (PoS), Byzantine Fault Tolerance (BFT), Proof of Authority (PoA) and Proof of Elapsed Time (PoET)
Unit III	BLOCKCHAIN-BASED FUTURES SYSTEM : Project presentation- Futures smart contract: Blockchain oracles- Web3j: Setting up the Web3J- Installing web3j- Wallet creation, Java client: The wrapper generator- Initializing web3j- Setting up Ethereum accounts- Deploying the contract
Unit IV	BLOCKCHAINS IN BUSINESS AND CREATING ICO : Public versus private and permissioned versus permission less blockchains- Privacy and anonymity in Ethereum- Why are privacy and anonymity important?
Unit V	- The Ethereum Enterprise Alliance- Blockchainas-a-Service- Initial Coin Offering (ICO): Project setup for ICO implementation- Token contracts- Token sale contracts-Contract security and testing the code.
Unit VI	DISTRIBUTED STORAGE IPFS AND SWARM : Ethereum Virtual Machine- Swarm and IPFS: Installing IPFS, Hosting our frontend: Serving your frontend using IFPS, Serving your frontend using Swarm, IPFS file uploader project: Project setup the web page
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	

Course Material/Learning Resources

Text books:

Imran Bashir, “Mastering Blockchain: Distributed Ledger Technology, decentralization, and smart contracts explained”, 2nd Edition, Packt Publishing Ltd, March 2018.

Bellaj Badr, Richard Horrocks, Xun (Brian) Wu, “Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger”, Packt Publishing Limited, 2018.

Reference Books:

6. Andreas M. Antonopoulos , “Mastering Bitcoin: Unlocking Digital Cryptocurrencies”, O’Reilly Media Inc, 2015

Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, “Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction”, Princeton University Press, 2016.

Weblink to Equivalent MOOC on SWAYAM if relevant:
https://nptel.ac.in/courses/106104220

Weblink to Equivalent Virtual Lab if relevant:
https://www.youtube.com/watch?v=YJyXfgBmc8

https://www.youtube.com/watch?v=SSo_EIwHSd4

Any pertinent media (recorded lectures, YouTube, etc.) if relevant:

Elective-I Code of the Course/Subject	Title of the Course/Subject	(Total Number of Hours)
4MCSW6(4)	**OOSE**	**60**

COs:
After Completion of this Course, students would be able to
1. **Understand Object Oriented Concepts and Modeling**
2. **Do basic and advanced structural modeling**
3. **Analyse Basic Behavioural Modeling**
4. **Analyse object oriented system design process**
5. **Perform object oriented analysis**
6. **Perform object oriented testing**

Unit	Content
Unit I	Object Oriented Concepts and Modeling : What is Object Orientation? Introduction to class, Object, inheritance, polymorphism), Model: Importance of Modeling, Object Oriented Modeling , Object oriented system development, Function/data methods, Object oriented analysis, Object oriented construction, Object oriented testing, Identifying the elements of an object mode, Identifying classes and objects, Specifying the attributes, Defining operations, Finalizing the object definition
Unit II	Introduction to UML, Overview of UML, Conceptual Model of UML, Architecture, S/W Development Life Cycle Basic and Advanced Structural Modeling: Classes Relationship, Common mechanism, Diagrams, Class diagram, Advanced classes, Advanced Relationship, Interface, Types and Roles, Packages, Object Diagram
Unit III	Basic Behavioral Modeling, Interactions, Use cases, Use Case Diagram, Interaction Diagram, Activity Diagram ,State chart Diagram Architectural Modeling, Component , Components Diagram, Deployment Diagram
Unit IV	Generic components of OO Design model, System Design process, Partitioning the analysis model, Concurrency and subsystem allocation, Task Mgmt component, Data Mgmt component, Resource Mgmt component, Inter sub-system communication, Object Design process
Unit V	Object Oriented Analysis, Iterative Development, Unified process & UP Phases Inception Elaboration Construction Transition, Understanding requirements, UP Disciplines, Agile UP
Unit VI	Object Oriented Testing, Overview of Testing and object oriented Testing, Types of Testing, Object oriented Testing strategies, Test case design for OO software, Inter class test case design
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity		

Course Material/Learning Resources

Text books:

1: The Unified Modeling Language User Guide by Grady Booch, James Raumbaugh, Ivar Jacobson.

2: Object Oriented Software Engineering by Ivar Jacobson

Reference Books:

1: Software Engineering by Pressman

2: Applying UML and Patterns by Craig Larman

Weblink to Equivalent MOOC on SWAYAM if relevant:

Elective-I Code of the Course/Subject	Title of the Course/Subject	(Total Number of Hours)
4MCSW6(5)	**Robotics and AI**	**60**

Course Objectives (Cos)

Upon successful completion of the course, students would be able to

1. **Understand Intelligent Agents**
2. **Analyse Search Algorithms and Expert Systems**
3. **Learn AI language (LISP)**
4. **Classify robots and apply robots in industrial application environment**
5. **Explain robot drives**
6. **Explain sensors and use machine vision**

Unit	Content
Unit I	Introduction to AI – Foundations of AI, Intelligent Agents – Agents and Environments – Good Behavior – Nature of Environments – Structure of Agents. Problem solving: Problem Solving Agents – Searching for solutions- Uninformed Search Strategies – Informed Search Strategies, heuristic functions.
Unit II	Search Algorithms : Local search algorithms and optimization problems – Searching with nondeterministic Actions, Constraint satisfaction problems. Expert systems – Introduction – Difference between expert system and conventional programs – Expert system organization – Architecture of Expert system – Knowledge representation techniques- Knowledge acquisition techniques - Inference Engine- Explanation systems.
Unit III	AI language: Lisp Lisps, Typing at Lisp, Defining Programs, Basic Flow of Control in Lisp, Lisp Style, Atoms and Lists, Basic Debugging, Building Up List Structure, More on Predicates, Properties, Pointers, Cell Notation and the Internals (Almost) of Lisp, Destructive Modification of Lists, The for Function ,Recursion, Scope of Variables, Input/ Output, Macros
Unit IV	Introduction to Robotics: Robot Anatomy - Coordinate Systems, Manipulators & Mobile Robots, Classification of Robots, Robot and effectors - special reference to servomotors Transmission and actuators, Robot Applications - Industrial application environment and work cells, Feeders and Oriented Device.
Unit V	Drive systems : Types of Robot Drives: Mechanical – Magnetic – Vacuum, Robot arm kinematics, World, Tool, DH transformation and Inverse Kinematics. Fundamentals of Closed loop control, PWM amplifiers, PID control.
Unit VI	Sensors and Machine Vision: Robotics sensors: Range, Proximity, Touch, Force and Torque Sensing, uses of sensors in Robotics, Applications- Inspection – Identification - Visual Serving and Navigation, Machine Vision - The sensing and digitizing function in Machine Vision - Image processing and analysis, Training and vision system, Robotic Application - Low and High-level vision.
***SEM** Assignment, Class test, Attendance, Seminar, Study tour, Industrial visit, Field work, Group discussion or any other innovative practice/activity	

Course Material/Learning Resources

Text books:

1. Stuart J Russell and Peter Norvig, Artificial Intelligence – A Modern Approach, PHI Learning, Third Edition, 2010.
2. Patterson W D, Introduction to Artificial Intelligence and Expert Systems, PHI Learning, First Edition, 1995.
3. Deb, S. R., Robotics Technology and Flexible Automation, Tata McGraw Hill publishing company limited, 1994.
4. Richard David Klafter, Thomas A. Chmielewski, Michael Negin, Robot Engineering: An integrated Approach, Prentice Hall of India Pvt. Ltd, 1994.

Reference Books:

1. Elaine Rich and Kelvin Knight, Artificial Intelligence, TMH, Third Edition, 2009.
2. King Sun Fu, Rafael C. González, C. S. George Lee, Robotics: control, sensing, vision, and intelligence, McGraw Hill, 1987.
3. Craig, J. J, Introduction to Robotics: Mechanics and Control, Addison-Wesley, London, Third Edition, 2004.
4. M.P.Groover, M. Weins, R.N.Nage, N.C.Odrey, Industrial Robotics, McGraw Hill
5. K.D. Richard, Chmielewski T.A and Michael, Robotic Engineering, PHI Learning.
6. K.S. Fu Gonzalez, Lee, Robotics Control, Sensing, Vision and intelligence.

Weblink to Equivalent MOOC on SWAYAM if relevant:

https://nptel.ac.in/courses/106105079

https://nptel.ac.in/courses/112104298

Weblink to Equivalent Virtual Lab if relevant:

Any pertinent media (recorded lectures, YouTube, etc.) if relevant:

Laboratories

Course Code	**4MCSW7**
Course Name	**Lab-VII 1, 2 – Testing and Data Analytics tools**

Total Credits	**2**
Course Outcomes	Course Outcome: Upon successful completion of the course, students would be able to • Get the basics of Software Testing • Run test cases for various applications. • Do the simple automation testings. • Use the basic HDFS commands • Use the Map Reduce programming for data analysis. • Work with Pig and Hive frameworks.

	Contents	**Total Hours**
	The sample list of programs is given below. This list can be used as a guideline for problem statements but the scope of the laboratory should not be limited to the same. Aim of the list is to inform about minimum expected outcomes. **Testing practical assignments** 1. Design black box test cases for windows notepad application 2. Design black box test cases for simple calculator application 3. Design black box test cases for web pages of chosen website 4. Identify the types of testing you would perform on ball pen to ensure the highest quality. 5. There is a simple program with the following items: a. Input Box A b. Input Box B c. ADD button d. Result Text Box [=A+B] Identify all the test cases for the program. [Example: press the Add button without entering anything in Input Box A and B] 6. Execute the test cases created for web pages of chosen website and find out the defects [Install and setup the test automation tool of your choice] 7. Create/Record a UI automation test case for windows notepad application 8. Create/Record a UI automation test case for a chosen website **Data Analysis Tool practical assignments** 1. Installing and configuring Hadoop 2. Practicals on File Management commands in HDFS 3. Word count Map Reduce Program to understand Map Reduce Paradigm. 4. Implementing Matrix Multiplication with Hadoop Map Reduce 5. Practical's on Pig Latin scripts to sort, group, join, project, and filter your data. 6. Practical's of Hive: Databases, Tables, Views, Functions and Indexes	

Course Code	**4MCSW8**
Course Name	**Lab-VIII 3, 4- IOT tools / Security Tools**
Total Credits	**2**
Course Outcomes	**Course Outcomes:** **Upon successful completion of the course, students would be able to** **1.** implement programs with Arduino efficiently. **2.** design hardware for IoT on different platforms for devices that can be connected to internet **3.** Acquire knowledge, thinking and designing capability, to solve real world engineering problems and analyse results. **4.** design components with realistic constraints, for IOT nodes and systems. **5.** implement various encryption techniques **6.** Construct codes for various authentication schemes **7.** Develop signature scheme using digital signature method **8.** Demonstrate various network security system using open source tools

	Contents	Total Hours

Practical's on IoT Tools

The sample list of programs is given below. This list can be used as a guideline for problem statements but the scope of the laboratory should not be limited to the same. Aim of the list is to inform about minimum expected outcomes. (Min 15)

1. To interface Push button/Digital sensor (IR/LDR) with Arduino and write a program to turn ON LED when push button is pressed or at sensor detection.
2. To interface LED/Buzzer with Arduino and write a program to turn ON LED for 1 sec after every 2 seconds.
3. To interface Bluetooth with Arduino and write a program to turn LED ON/OFF when '1'/'0' is received from a smartphone using Bluetooth.
4. To interface Bluetooth with Arduino and write a program to turn LED ON/OFF when 170 is received from a smartphone using Bluetooth.
5. Write a program on Raspberry Pi to retrieve temperature and humidity data from thingspeak cloud.
6. To interface the DHT11 sensor with Raspberry Pi and write a program to print temperature and humidity readings.
7. To interface the motor using a relay with Raspberry Pi and write a program to turn ON the motor when push button is pressed.
8. To insert MySQL database on Raspberry Pi and perform RC basic SQL queries
9. Write a program on Arduino/Raspberry Pi to publish temperature data to MOTT Broker.
10. Install MySQL database on Raspberry Pi and perform basic SQL queries.
11. Implementation of Substitution and Transposition ciphers
12. Implementation of Data Encryption Standard
13. Implementation of International Data Encryption Algorithm
14. Implementation of Advanced Encryption Standard
15. Implementation of RSA Algorithm
16. Implementation of Diffie-Hellman Key Exchange
17. Implementation of Message Authentication Codes
18. Implementation of Hash functions
19. Implementation of Digital Signature Standard
20. Hiding of confidential information within Image
21. Implementation in FOSS based security mechanisms