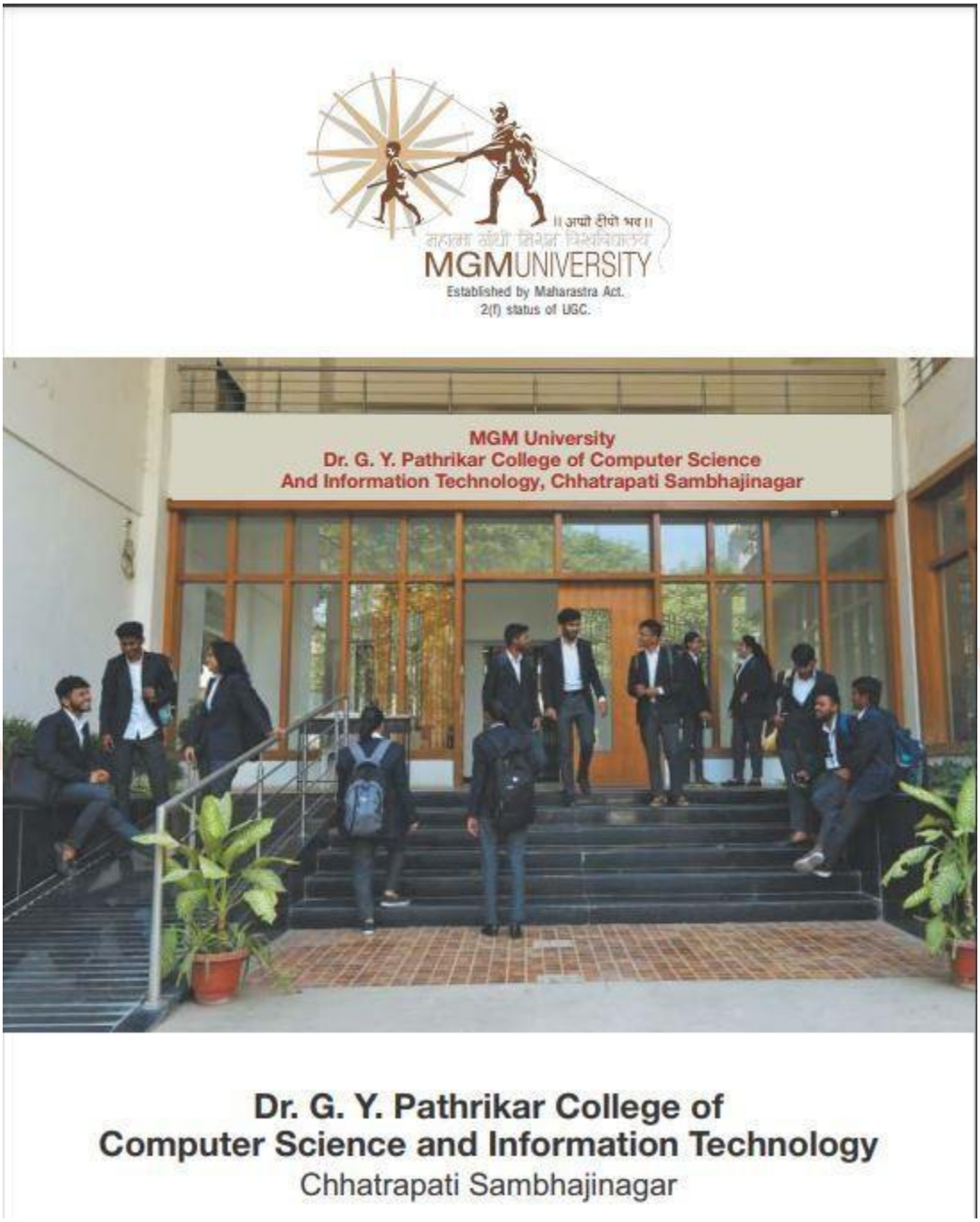

B.C.A. (Hons. /Hons. with Research) Science:

First, Second, Third & Fourth Year (Syllabus)

With Effect From: ACADEMIC YEAR: 2023-24

MGM University

Vision

- To ensure sustainable human development which encourages self-reliant and self-content society.
- To promote activities related to community services, social welfare and also Indian heritage and culture.
- To inculcate the culture of non-violence and truthfulness through vipassanna meditation and Gandhian Philosophy.
- To develop the culture of simple living and high thinking

Mission

- To impart state of art education and technical expertise to students and give necessary training to teachers to create self-reliant society for future.
- To encourage students to participate in Indian and International activities in sports, literature, etc. so that future generation becomes base for free and liberal society
- To educate students in areas like Management, Finance, Human relations to inculcate philosophy of simple living and high thinking value of simple economic society.
- To inculcate culture of non-violence and truthfulness through Vipassana.

  To sustain activities of Indian culture (viz. classical dance, music and fine arts) through establishing institutes like Mahagami, Naturopathy, etc.

विद्यापीठ गीत

अत्त दिप भव भव प्रदिप भव,

स्वरूप रूप भव हो

ज्ञान सब्ब विज्ञान सब्ब भव ,

सब्ब दिप भव हो

अत्ताहि अत्त नो नाथो ,

अत्ताहि अत्त नो गति

अत्त मार्गपर अप्रमादसे है तुझे चलना

सब्ब का कल्याण हो ,

वो कार्यकुशल करना

सब्ब का उत्तम मंगल , पथप्रदर्शक हो

अत्त दिप भव भव प्रदिप भव ,

स्वरूप रूप भव हो

ज्ञान सब्ब विज्ञान सब्ब भव ,

सब्ब दिप भव हो

बुद्धमं शरनं गच्छामि :

धम्मं शरनं गच्छामि :

संघं शरनं गच्छामि :

Dr. G. Y. Pathrikar College of Computer Science & Information Technology

MGM college of Computer Science and Information Technology was established in 2001 offering undergraduate and postgraduate degree program in Computer Science and Information Technology. College was renamed as Dr.G.Y.Pathrikar College of Computer Science and Information Technology in 2003 in memory of great educationalist, one of the founder member and Ex-Secretary MGM, Dr.G.Y.Pathrikar Sir.

It is first self-financed ISO certified institution offering program dedicated to Computer science and Information technology in Maharashtra and has achieved status of 2f/12b. Ours was the only and first college to be re-accredited as A+ grade with NAAC in the year 2017. Experienced and qualified faculty with Ph.D is strength of our college. Starting with 77 student's College has crossed total students strength of 10,000 passing out. Student are doing well in various MNCs like Infosys, Tech-Mahindra, Wipro, Capgemini, Cognizant etc. Many have their own Startups. Some of the students have completed their Masters and Ph.D. program from foreign countries like US, UK, Australia. Now we are constituent college of MGM University, Chhatrapati Sambhajinagar.

Vision

To be an academic institution in dynamic equilibrium in social, ecological and economical environment striving continuously for excellence in total quality education, research and technological service to the nation.

Mission

- To create and sustain a community of learning in which students acquire knowledge and learn to apply it professionally with due consideration for ethical, and economical issues.
- To upgrade our students in all respect with the help of latest infrastructure in the area of Computer Science and Information Technology in order to build the National Capabilities.
- To understand the culture of Non-violance, truth, peace through Gandhian Philosophy.

Programs offered at Dr. G. Y. Pathrikar College of Computer Science & Information Technology

Undergraduate Programmes	Postgraduate Programmes	PhD Programmes
B.Sc(Computer Science) Honours / Honours with Research	M.Sc(Computer Science)	Ph.D. in Computer Science and Information Technology
B.Sc(Information Technology) Honours/ Honours with Research	M.Sc(Information Technology)	
BCA(Science) Honours / Honours with Research	M.Sc(Data Science)	
B.Sc(Animation) Honours / Honours with Research	M.Sc(Animation)	
Integrated M.Sc. Data Science		
BCA(Digital Marketing) Honours		
B.Sc(Robotics) Honours		

Name of Faculty: Basic and Applied Science
Name of the College/Institute/Department/School: Dr. G.Y. Pathrikar College of CS& IT
Name of the Programme: B.C.A. (Science) Honours
Programme Type (UG/PG): UG
Duration: 4 Years

List of Options to select from Bucket of Courses provided in various categories:

Major	
Computer Application	
Core Major	**Core Elective**

Minor options for basic and applied science Faculty	**GYP**	**IBT**	**UDBAS**
	Cyber Security	Food Technology and Processing	Chemistry
	Robotics	Microbiology	Geo-Informatics
	Data Analytics	Biotechnology	Mathematics
		Bioinformatics	Statistics
		Food Nutrition and Dietetics	Material Science

Minor options from Other Faculty	**Faculty of Engineering and Technology**	**Faculty of Social Sciences & Humanities**	**Faculty of Design**	**Faculty of Management and Commerce**	**Interdisciplinary Faculty**	**Performing Arts**
	Data Science	Filmmaking	Product Design	Financial Management	Cosmetic Technology	Theatre Arts
	IoT	Photography	Interior Design	E-Commerce	Education	Dance
	Geo-informatics and Applications	Mass Communication and Journalism	Contemporary Arts	International Business Management	Yog Sciences	Music
	EV Technology	Psychology	Visual Communication	Hospitality Mgmt	Physical Education	Folk Art
	Drone Technology	Economics	Fashion Technology	Travel and Tourism	Home Science	
	Robotics Technology	English		Art of Leadership		
	Chemical Technology	Social Work		Art of Business		
	AI&ML					
	Universal Human Values					
	Energy management					

Name of the Programme: BCA (Computer Application) Honours with Research
Programme Type (UG/PG): UG
Duration: Four Years

First Year - Semester I												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML101	Computer Architecture	Lecture	2	2		30	20	50	-		20
MM	CAS41MML102	C Programming	Lecture	2	2		30	20	50	-	08	20
MM	CAS41MMP101	Practical based on Computer System Architecture	Practical	1		2	30	20	50	-	08	20
MM	CAS41MMP102	Practical based on C Programming	Practical	1		2	30	20	50	-	08	20
IKS	CAS41IKL101	Indian Psychology and yoga	Lecture	2	2	-	30	20	50	-	08	20
AEC		Basket of AEC From University	Lecture	2	2	-	30	20	50	-		20
OE		Basket of OE From University	Lecture	2	2	-	30	20	50	-		20
OE		Basket of OE From University	Lecture	2	2	-	30	20	50	-	08	20
VSC	CAS41VSP101	LINUX Operating System	Practical	2		4	30	20	50	-	08	20
SEC	CAS41SEL101	Discrete Mathematics	Lecture	2	2	-	30	20	50	-		20
VEC		Basket of VEC From University	Lecture	2	2	-	30	20	50	-	08	20
CC		Basket of CC From University	Practical	2	-	4	50	-	50	20		20
Total				**22**	**16**	**12**	**380**	**220**	**600**			

Note:

Nature of Course : L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation, **Course Category: M**M-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

First Year - Semester II

Course Category	Course Code	Course Title	Nature of Course	No. of Credits	Teaching (Contact hrs/ week)		Evaluation Scheme (Marks)			Minimum Passing (Marks)		
					L	P	Internal	External	Total	Internal	External	
MM	CAS41MML103	Linear Data Structure	Lecture	2	2		30	20	50		08	20
MM	CAS41MML104	Advance C Programming	Lecture	2	2		30	20	50		08	20
MM	CAS41MMP103	Practical based on Linear Data Structure	Practical	1		2	30	20	50		08	20
MM	CAS41MMP104	Practical based on Advance C Programming	Practical	1		2	30	20	50		08	
MI		Basket of MI From University	Lecture	2	2	-	30	20	50		08	20
AEC		Basket of AEC From University	Lecture	2	2	-	30	20	50		08	20
OE		Basket of OE From University	Lecture	2	2	-	30	20	50		08	
OE		Basket of OE From University	Lecture	2	2	-	30	20	50		08	20
VSC	CAS41VSP102	Structural Query Language	Practical	2		4	30	20	50		08	20
SEC	CAS41SEL102	Data Base Management System	Lecture	2	2	-	30	20	50		08	20
VEC		Basket of VEC From University	Lecture	2	2	-	30	20	50		08	
CC		Basket of CC From University	Practical	2	-	4	50	-	50	20	-	
Total				**22**	**16**	**12**	**380**	**220**	**600**			

Note:

Nature of Course : L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Second Year - Semester III												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS4MML201	Non Linear Data Structure	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML202	Object Oriented Programming (C++)	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML203	Fundamental of Computer Network	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MMP201	Practical Based on Non Linear Data Structure	Practical	1	-	2	30	20	50		08	20
MM	CAS4MMP202	Practical Based on Object Oriented Programming (C++)	Practical	1	-	2	30	20	50		08	20
OE		Basket of OE From University	Lecture	2	2	-	30	20	50		08	20
MI		Basket of MI From University	Lecture	3	3	-	60	40	100		16	40
MI		Basket of MI From University	Practical	1	-	2	30	20	50		08	20
AEC		Basket of AEC From University	Lecture	2	2	-	30	20	50		08	20
VSC	CAS41VSP201	Advance Excel	Practical	2		4	30	20	50		08	20
FP	CAS41FPJ201	Field Project	Project	2	-	4	50	-	50	20	-	20
CC		Basket of CC From University	Practical	2	-	4	50	-	50	20	-	20
Total				**22**	**13**	**18**	**430**	**220**	**650**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project

Second Year - Semester IV												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML204	Advance Database Management System	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML205	Web Technologies	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML206	Advance Computer Network	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MMP203	Practical Based on Advance Database Management System	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP204	Practical Based on Web Technologies	Practical	1	-	2	30	20	50		08	20
OE		Basket of OE From University	Lecture	2	2	-	30	20	50		08	20
MI		Basket of MI From University	Lecture	3	3	-	60	40	100		16	40
MI		Basket of MI From University	Practical	1	-	2	30	20	50		08	20
AEC		Basket of AEC From University	Lecture	2	2	-	30	20	50		08	20
SEC	CAS41SEP 201	PHP	Practical	2		4	30	20	50		08	20
CEP	CAS41CEP 201	Community Engagement Program(As Per University Guidelines)	Practical	2	-	4	50	-	50	20	-	20
CC		Basket of CC From University	Practical	2	-	4	50	-	50	20	-	20
Total				**22**	**13**	**18**	**430**	**220**	**650**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Second Year - Semester V												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML301	Software Project Management	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML302	Core Java	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML303	Data Science	Lecture	2	2		30	20	50		08	
MM	CAS41MMP301	Practical Based on Software Project Management	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP302	Practical Based on Core Java	Practical	1	-	2	30	20	50		08	20
ME	CAS41MEL301	Multidimensional Computer Graphics	Lecture	3	3	-	60	40	100		16	40
	CAS41MEL302	Advance PHP for Content Management System										
ME	CAS41MEP301	Pr. Based on Multidimensional Computer Graphics	Practical	1	-	2	30	20	50		08	20
	CAS41MEP302	Pr. Based on Advance PHP for Content Management System										
MI		Basket of MI From University	Lecture	3	3	-	60	40	100		16	40
MI		Basket of MI From University	Practical	1	-	2	30	20	50		08	20
VSC	CAS41VSP301	Android Application Development	Practical	2	-	4	30	20	50		08	20
FP	CAS41FPJ301	Field Project	Project	2	-	4	50	-	50	20	-	20
Total				**20**	**12**	**16**	**410**	**240**	**650**			**280**

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Second Year - Semester VI												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML304	Software Project & Agile Development	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML305	Advance JAVA	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MML306	Network Security	Lecture	2	2	-	30	20	50		08	20
MM	CAS41MMP303	Practical Based on Software Project & Agile Development	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP304	Practical Based on Advance JAVA	Practical	1	-	2	30	20	50		08	20
ME	CAS41MEL303	AWS DevOps	Lecture	3	3	-	60	40	100		16	40
	CAS41MEL304	Data Mining and Visualization										
ME	CAS41MEP303	Practical Based on AWS DevOps	Practical	1	-	2	30	20	50		08	20
	CAS41MEP304	Practical Based on Data Mining and Visualization										
MI		Basket of MI From University	Lecture	3	3	-	60	40	100		16	40
MI		Basket of MI From University	Practical	1	-	2	30	20	50		08	20
OJT	CAS41JTP301	On Job Training	Practical	4		8	60	40	100		16	40
Total				**20**	**12**	**16**	**390**	**260**	**650**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Fourth Year- Semester VII												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML401	Software Testing and Quality Assurance	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MML402	BlockChain Technology	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MML403	Python Programming	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MMP401	Practical Based on Software Testing and Quality Assurance	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP402	Practical Based on BlockChain Technology	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP403	Practical Based on Python Programming	Practical	1	-	2	30	20	50		08	20
ME	CAS41MEL401	React JAVA Script	Lecture	3	3	-	60	40	100		16	40
	CAS41MEL402	Frontend Development (ASP.Net,Angular)					60	40	100		16	40
ME	CAS41MEP401	Practical Based on React JAVA Script	Practical	1	-	2	30	20	50		08	20
	CAS41MEP402	Practical Based on Frontend Development (ASP.Net, Angular)					30	20	50		08	20
RM	CAS41RML401	Research Methodology	Lecture	3	3	-	60	40	100		16	40
RM	CAS41RMP401	Practical based on Research Methodology	Practical	1	-	2	30	20	50		08	20
Total				**20**	**15**	**10**	**450**	**300**	**750**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Fourth Year- Semester VIII												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML404	Machine Learning with Python	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MML405	Block chain Platforms	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MML406	Biomedical Image Processing	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MMP404	Practical Based on Machine Learning with Python	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP405	Practical Based on Block chain Platforms	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP406	Practical Based on Biomedical Image Processing	Practical	1	-	2	30	20	50		08	20
ME	CAS41MEL403	Grid & Cloud Computing	Lecture	3	3	-	60	40	100		16	40
	CAS41MEL404	Big Data Analytics					60	40	100		16	40
ME	CAS41MEP403	Practical Based on Grid & Cloud Computing	Practical	1		2	30	20	50		08	20
	CAS41MEP404	Practical Based on Big Data Analytics					30	20	50		08	20
OJT	CAS41JTP401	On job Training	Practical	4		8	60	40	100		16	40
Total				**20**	**12**	**16**	**420**	**280**	**700**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Fourth Year- Semester VII

Course Category	Course Code	Course Title	Nature of Course	No. of Credits	Teaching (Contact hrs/ week)		Evaluation Scheme (Marks)			Minimum Passing (Marks)		
					L	P	Internal	External	Total	Internal	External	Total
MM	CAS41MML407	Geospatial Information System	Lecture	3	3	-	60	40	100	16		40
MM	CAS41MML408	Signal Processing	Lecture	3	1	-	60	40	100	16		40
MM	CAS41MMP406	Practical Based on Geospatial Information System	Practical	1	1	2	30	20	50		08	20
MM	CAS41MMP407	Practical Based on Signal Processing	Practical	1	3	2	30	20	50		08	20
ME	CAS41MEL405	Graphical User Interface	Lecture	3	1	-	60	40	100	16		40
	CAS41MEL406	Biometric technology										
ME	CAS41MEP405	Practical Based on Graphical User Interface	Practical	1	3	2	30	20	50		08	20
	CAS41MEP406	Practical Based on Biometric technology										
RM	CAS41RML401	Research Methodology	Lecture	3	1	-	60	40	100	16		40
	CAS41RMP401	Practical based on Research Methodology	Practical	1	4	2	30	20	50			20
RP	CAS41RPJ401	Research Project	Practical	4	20	8	60	40	100		08	40
Total				**20**	**12**	**16**	**420**	**280**	**700**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Fourth Year- Semester VIII												
Course Category	**Course Code**	**Course Title**	**Nature of Course**	**No. of Credits**	**Teaching (Contact hrs/ week)**		**Evaluation Scheme (Marks)**			**Minimum Passing (Marks)**		
					L	**P**	**Internal**	**External**	**Total**	**Internal**	**External**	**Total**
MM	CAS41MML408	Human Computer Interface	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MML409	Biomedical Image Processing	Lecture	3	3	-	60	40	100		16	40
MM	CAS41MMP408	Practical Based on Human Computer Interface	Practical	1	-	2	30	20	50		08	20
MM	CAS41MMP409	Practical Based on Biomedical Image Processing	Practical	1	-	2	30	20	50		08	20
ME	CAS41MEL407	Artificial Intelligence	Lecture	3	3	-	60	40	100		16	40
	CAS41MEL408	Big Data Analytics										
ME	CAS41MEP407	Practical Based on Artificial Intelligence	Practical	1	-	2	30	20	50		08	20
	CAS41MEP408	Practical Based on Big Data Analytics										
RP	CAS41RPJ402	Research Project	Practical	8		16	120	80	200		32	80
Total				**20**	**09**	**22**	**390**	**260**	**650**			

Note:

Nature of Course: L- Lecture, P-Practical, S-Seminar, J-Project, I-Internship, D-Dissertation

Course Category: MM-Major Mandatory, ME-Major Elective, MI-Minor, OE-Generic / Open electives, VSC-Vocational skill course, SEC-Skill Enhancement course, AEC-Ability Enhancement course, IKS-Indian Knowledge system, VEC-Value Education course, OJT-On Job Training / Internship / Apprenticeship, FP-Field project, CEP-Community engagement and service, CC-Co – curricular course, RM-Research methodology, RP-Research project.

Semester: FIRST

Syllabus
Semester – I

Course code: CAS41MML101 **Course name:** Computer Architecture **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.
Course Objectives: Student understands the basic structure and operation of a digital computer and various processors.
Course Outcomes: After completion of the course the student will be able to:
CO1: Student understands of the basic structure and operation of a digital computer.
CO2: To learn the architecture and assembly language Programming of microprocessor.
CO3: To study the different Processors.

Contents

Unit	Content	Teaching hours
1	**Computer Arithmetic:** Number System: Decimal System Binary Number System, Hexadecimal Number System,Octal Number System**,**Number Conversion: Decimal to Other, Binary to Other, Octal to Other, Hexadecimal to Other, BCD Numbers, ASCII Code, Computer Arithmetic: Addition, Subtraction. **Logic Gates & Boolean Algebra**: Positive & Negative Logic, Truth Table, Logic Gates: AND, OR, NOT, NAND, NOR and Exclusive OR Gate, Universal Gates. Postulates &Theorems of Boolean Algebra(Idempotent, Complementation, Commutative, Associative, Distributive, De-Morgan's Theorem)	10
2	**Arithmetic Circuits:** Combinational Circuits, Implementing Combinational logic, Arithmetic Circuits: Half-Adder, Full-Adder, Half-Sub tractor, Full-Sub tractor. **Fundamentals of Microprocessors**: Comparison of 8-bit, 16-bitand32-bit microprocessor, 8086 Pin Configuration,8086Internal Architectures, Execution Unit & Bus Interface, Flag Registers, Introduction to Addressing Modes	10
3	**8086Interrupt and Interrupt Applications:** Interrupts of 8086, Hardware Interrupts, Software Interrupts. **Latest Trends in Microprocessor:** RISC and CISC Architectures, Design: Multicore Processor and Multicore Processing, Multicore Technology and Intel, Dual Core and Core Duo Processors, Corei3,i5,Mobile Processors	10

Text Books:
1.Anil K. Maini ,"Digital Electronics: Principles, Devices and Applications", Wiley Publication
2.Lyla B Das,"Micro Processors & Multi core systems", Pearson Publication
Reference Books:
1.Douglas V Hall ,"Microprocessor and Interfacing", Tata McGraw Hill
2.M. Morris Mano,"Digital Design".
WebsiteResource:https://www.javatpoint.com/computer-organization-and-architecture- tutorial

Semester I

Course code: CAS41MMP101 **Course name:** Practical Based on Computer Architecture **Course category:** Major Mandatory	
Credits: 1	**Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.	
Course Objectives: Student understands of the basic structure and operation of a digital computer and various processors	
Course Outcomes: At the end of the course, the students will be able to -	
CO1: Student understands of the basic structure and operation of a digital computer.	
CO2: To learn the architecture and assembly language Programming of microprocessor.	
CO3: To study the different Processors.	

Contents

Unit	Content	Teaching hours
1	To Perform Number systems Conversations, To Perform Binary Arithmetic operations, To Verify the truth table of Basic Logic Gates, To Verify the truth table of Universal Logic Gates, To verify the truth table of Special Purpose Logic Gates.	04
2	State and Prove Demorgan's Theorem, To Study and Verify Combinational Logic Circuits (Half adder), To Study and Verify Combinational Logic Circuits (Fulladder), To Study General Purpose Registers of 8086Microprocessor,To Study Special Purpose Registers of 8086 Microprocessor.	04
3	To Study 8086 Pin Diagram and its Functioning, To Study 8086 Interrupt and its Applications, To Study Pentium Pro Architecture, To Analyze and Compare Pentium and Core-i3 Processor, To Analyze and Compare RISC and CISC Architecture	04

Text Books:
1.AnilK.Maini,"Digital Electronics: Principles, Devices and Applications", WileyPublication
2.LylaBDas,"MicroProcessors & Multi core systems", Pearson Publication
Reference Books:
1.DouglasVHall,"Microprocessor and Interfacing",Tata McGraw Hill
Website Resources:https://www.javatpoint.com/computer-organization-and-architecture-tutorial

Semester I

Course code: CAS41MML102 **Course name:** Introduction to C Programming **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of computer application.
Course Objectives: To introduce the foundations of computing, programming and problem- solving using C programming language basics.
Course Outcomes: At the end of the course, the students will be able to -
CO1: It aims to train the student to the basic concepts of the C programming language. This course involves a lab component which is designed to give the student hands-on experience with the concepts.
CO2: The course aims to provide exposure to problem-solving and principles through programming.
CO3: Write the C code for a given algorithm
CO4: Identify tasks in which the numerical techniques learned are applicable and apply them to write programs, and hence use computers effectively to solve the task

Contents

Unit	Content	Teaching hours
1	**Introduction:** An Overview of C, History of C language, Features of C. **Basic Elements & Operators:** Character set, C Token, Identifier & Keywords, Variables**. Constant and its types**: Integer constant, floating point constant, character constant, string constants. **Operators:** Binary Operator**:** Arithmetic, Relational, And Logical, Unary operators: Increment & decrement, Assignment and Conditional operator. **Data Types:** Data Types: *int, char, float, double*. Declaration & Initialization.	10
2	**C Program & I/O statements:**Structure of C Program,Compilation & Execution of C program**,**I/O: Introduction, Formatted Input/Output function: *scanf()&printf()*, Escape sequence characters.**Control and Iterative Statements:**Simple if, nested if, if-else, else if ladder**,**Switch-case statement**,**The conditional expression (? : operator)**, Loops:** while and do-while loop, and for loop , break & continue statement, *goto*statement.	10
3	**Arrays:** Introduction, Declaration and initialization **,**Access in array elements,Memory representation of array,One dimension and multi-dimensional arrays **Character array**, Introduction to string.**String handling functions**: strcpy(), strcmp(), strcat(),strlen(),strupr(),strlwr(),gets(),puts()	10

Text Books:
1.Y.P.Kanetkar ,"Letus C",B pbpublication
2.E.Balaburuswamy,"Programming in C", TataMacgrawHill
Reference Books:
1.Goterfried,"Programming in C",Shaums'Series
Website Resources:https://www.w3schools.com/c/ **https://www.javatpoint.com/c-programming-language-tutorial** https://www.geeksforgeeks.org/c-programming-language/

Semester I

Course code: CAS41MMP102 **Course name:** Practical Based on Introduction to C Programming **Course category:** Major Mandatory
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of computer application.
Course Objectives: To make students understand about the practical implementation of C programs
Course Outcomes: After completion of the course the student will be able to:
CO1: It aims to train the student to the basic concepts of the C programming language.
CO2: To understand about syntax of all the basic structures of C programming language.
CO3: Learn to build programs based on various concept to solve real life problems

Contents

Unit	Content	Teaching hours
1	C "Hello, World!" Program,C Program to Print an Integer (Entered by the User),C Program to Add Two Integers(Use of operators),C Program to Multiply Two Floating-Point Numbers,C Program to Compute Quotient and Remainder	04
2	C Program to Check Whether a Number is Even or Odd,C Program to Check Whether a Character is a Vowel or Consonant,C Program to Find the Largest Number Among Three Numbers,C Program to Find Factorial of a Number,C Program to Calculate the Power of a Number	04
3	C Program to Calculate Average Using Arrays,C Program to Find Largest Element in an Array,C Program to Add Two Matrices Using Multi-dimensional Arrays,C Program to Find Transpose of a Matrix,C program to demonstrate various string handling functions	04

Text Books:
1.Y.P.Kanetkar ,"Letus C", B pbpublication
2.E.Balaburuswamy,"Programming in C", TataMacgrawHill
Reference Books:
1.Goterfried,"Programming in C", Shaums'Series

Semester I

Course code: CAS41VSP101 **Course name:** LINUX Operating System **Course category:** Vocational Skill Course
Credits :2 **Teaching scheme:** L-0 P-4 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of operating system functionality
Course Objectives: Introducemodernoperatingsystemsbasicconceptsandcommandstoworkwith.
Course Outcomes: After completion of the course the student will be able to:
CO1: Understand and make use of linux command for working with shell
CO2: Understand, apply utilities commands that are essential while working with linux operating system.
CO3: Gain knowledge of linux file system and associated commands.

Contents

Unit	Content	Teaching hours
1	**WORKINGWITH SHELL:** Identify the running shell(related commands)**,**Correcting mistakes(related commands)**,**Repeating/Editing command line(related commands)**,**Finding documentation (related commands)**,**Getting help with the system(related commands)	04
2	**UITILTIES :** Basic utilities(List the names of file, display at ext file, delete a file, display a text file one screen at a time, display system name)**,**Working with files(commands associated with files)**,**Compressing and archiving files**,** Locating commands**,**Obtaining user information, system information**,,**Communicating with other users	04
3	**FILE SYSTEM :**Working with directories (commands associated with directory)**,**Access permissions (commands associated with access permissions)**,**Access Control Lists (commands associated with ACLs) **,**Hard links(commands associated with hard links)**,**Symbolic links (commands associated with symbolic links)	04

Text Books :
1.Mark G. Sobel, ”A Practical Guide to Linux Commands, Editors, and Shell Programming”.
2.Adam Vardy,” Linux For Beginners: The Ultimate Guide To The Linux Operating System & Linux”.
Text Books :
1.Ted Dawson,” Linux: The Ultimate Step by Step Guide to Quickly and Easily Learning Linux”.
2.Greg Tomsho ,” Guide to Operating Systems “.

Semester I

Course code: CAS41SEL101 **Course name:** Discrete Mathematics **Course category:** Skill Enhancement Course
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of mathematics
Course Objectives: Capable to understand Combinations, Propositional function, statements and well-formed formulas; to understand and handle the concept of Set theory; able to understand the concept of Relations and its types; to understand various concepts in graphs and trees
Course Outcomes: After completion of the course the student will be able to:
CO1: To understand the representation of various statements using set, relations, functions, permutations and combinations, groups, graphs and trees.
CO2: Use logical notations to formulate and reason about fundamental Mathematical concepts such as sets, relations, functions and algebraic structures
CO3: Analyse the growth of functions and real world problems using various concepts like recurrence relations, graph implementation etc.

Contents

Unit	Content	Teaching hours
1	**Set Theory:**Types of Set: Finite, Infinite ,Singleton ,Empty, Subset,Proper Subset, Universal Set ,Power Set,Venn Diagram,Operations on Set: Union of Sets, Intersection of Sets,Complement of Set,Cartesian Product ,Difference and Symmetric Difference of SetPrinciple of Inclusion and Exclusion for two set,Principle of Inclusion and Exclusion for three sets **Combinatory:**Permutation and Combination**,**Mathematical Induction-Pigeonhole principle	10
2	**Relations:**Basic definitions of Relation **,**Types of Relations **,**Graph of Relations **,**Properties of Binary Relations**,**Matrix Representation of Relations**,** Operations on Relations**,**Partition and Covering**,**Transitive Closure **,**Partial Ordering Relations.	10
3	**Basics of Graph Theory and Tree:** Introduction to Graph**,**Application of Graph**,**Finite and Infinite Graph**,**Incidence and Degree**,**Null Graph**,**Isolated and Pendent Vertex, Isomorphism,Union and Intersection Operations on Graph,Subgraph, Planner Graph,Walks, Path and Circuit Introduction to Trees,Pendant Vertices on Tree,Binary Tree**,**Spanning Tree	10

Text Books:
1.C. L Liu, D.P.Mohapatra,"Elements of Discrete Mathematics-A Computer Oriented Approach",TataMcGrawHill
2.K.H.Rosen,"Discrete Mathematics and its Applications With Combinatorics and Graph Theory",TataMcGrawHill
Reference Books:
1.J.P.Tremblay,R.Manohar,"Discrete Mathematical Structures with Applications to Computer Science",TataMcGrawHill, India
2.BernandKolman, Roberty C.Busby,SharnCutterRoss,"Discrete Mathematical Structures",PearsonEducation/PHI

Semester: SECOND

Semester II

Course code: CAS41MML103 **Course name:** Linear Data Structures **Course category:** Major Mandatory	
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20	
Pre-requisites: Basic of computer Programming and aware about data.	
Course Objectives: Student get familiar with the basic concepts of data structures and algorithms, Get familiar with basic techniques of algorithms. Student get familiar with basic concepts about stacks, queues, lists, and hence student able to implement practically searching techniques.	
Course Outcomes: After completion of the course the student will be able to:	
CO1: Ability to implement and use linear data structures, including stacks, queues, lists.	
CO2: Understand of basic terminology data, data item, fields and data structures.	
CO3: Understand and analyze the concepts of arrays.	

Contents

Unit	Content	Teaching hours
1	**Introduction Design of Algorithm in Data Structure:** Definition of data and meaning of Algorithm ,Definition of data structure, classification and types of data structure, Basic Terminology: Data item, Fields, Records, Files, Entity, and Attributes. **Arrays :**Representation of Linear Arrays **,**Operations on Array-Traversing, Insertion and Deletions,Searching an element **,**MultidimensionalArrays:2D& M-D Concept.	10
2	**Stack:** Working of stack **,**Operations on stack **,**Array Representation of stack, Linked representation of stack, Algorithm for Insertion and deletion of an element**,** Searching an element**,** Application of stacks. **Queue:** Working of queue **,**Operations on queue **,**Representation of queues & link.Algorithm for insertion and deletion of an elemen,Searching an element,Dequeue: representation, insertion and deletion of an element Priority Queue: representation, insertion and deletion of an element.	10
3	**Linked Li:** Concept of linked list **,**Representation of linked list in memory **,**Traversing a linked list ,Insertion of an element ,Deletion of an element in linked list Types of linked list Header Linked List, Two way Linked List	10

Text Books :
1.Horowitz & Sahani ,"Fundamentals of Data structures",Galgotia publication Galgotia publication
2.Thomas H.Cormen, Charles E.Leiserson, Ronaild L.Rivest, CliffordStein ,"Introduction to Algorithms ,PHIPublication
Reference Books:
1.Tannenbaum,"Data Structures",PHIpublication

Semester II

Course code: CAS41MMP10 **Course name:** Practical based on Linear Data Structures **Course category:** Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.
Course Objectives: Practically, student get familiar with the basic concepts of data structures and algorithms, Get familiar with basic techniques of algorithms. Student get familiar with basic concepts about stacks, queues, lists, and hence student able to implement practically searching techniques.
Course Outcomes: At the end of the course, the students will be able to –
CO1: Ability to implement and use linear data structures, including stacks, queues, lists.
CO2: Understand of basic terminology data, data item, fields and data structures
CO3: Understand and analyse the concepts of arrays
CO4: Apply the concepts of linked list, Linked representation of Queue for specified applications.

Contents

Unit	Content	Teaching hours
1	**Introduction Design of Algorithm in Data Structure & Arrays:** W. P. in C to calculate the no. of letters in a word,W. P. in C to calculate the no. of words in a sentence,Implement algorithm for Traversing of Linear Array,Implement algorithm for insert new element in to Array,Implement algorithm for delete element from the Array.	04
2	**Stack and Queue:**Write a Program in C to find Prime numbers between 1 to1000numbers ,Implement algorithm for Traversing of Stack,Implement algorithm for PUSH new element into stack by algorithm,mplement algorithm for POP element from the Stack by algorithm,Implement algorithm for linear Search by algorithm.	04
3	**Linked List:** Implement algorithm for Traversing of Linked List,Implement algorithm for insert new element into Array,Implement algorithm for delete element from the Arra,Write a Program in C for interchange the place word1 with word2,Implement algorithm for traversing of Header Linked List or Two way Linked List	04

Text Books:
1.Horowitz & Sahani ," Fundamentals of Data structures ", Galgotia publication
2.Thomas H. Cormen, Charles E. Leiserson, Ronaild L.Rivest, CliffordStein ," Introduction to Algorithms ",PHIPublication
Reference Books:
1.Tannenbaum ," Data Structures ",PHIPublication

Semester II

Course code: CAS41MML104 **Course name:** Advanced C Programming **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.
Course Objectives: Practically, student get familiar with the basic concepts of data structures and algorithms, Get familiar with basic techniques of algorithms. Student get familiar with basic concepts about stacks, queues, lists, and hence student able to implement practically searching techniques.
Course Outcomes: At the end of the course, the students will be able to -
CO1: The course aims to provide exposure to problem-solving and principles through programming.
CO2: It aims to train the student to the basic concepts of the C programming language. This course involves a lab component which is designed to give the student hands-on experience with the concepts
CO3: Write the C code for a given algorithm.
CO4: Identify tasks in which the numerical techniques learned are applicable and apply them to write programs, and hence use computers effectively to solve the task

Contents

Unit	Content	Teaching hours
1	**Functions:**Introduction, Defining functions, Arguments, Function prototype, actual parameters and formal parameters, Calling function, Returning function results,Types of functions, Function with no argument and no return type, Function with argument and no return type, Function with no argument and return type, Function with argument and return type Recursion .**Preprocessor Directives:**File inclusion and conditional compiler directives, Macrosubstitution, #define, #if, #ifdef, #else, #elif, #endif.	10
2	**Structure & Union** **Structure**: Introduction, Declaration and initializing structure, accessing structure members, Nested structures, Arrays of structure, *type def* statement. **Unions**: Declaration, Difference between structure and union **Pointers:**Introduction, Memory organization. Declaration and initialization of pointers. The pointer operator * and &, De-referencing, Pointer expression and pointer arithmetic	10
3	**File Handling:**Introduction, Opening & closing a file, Input/Output operations on files, text and binary files, getc() ,putc() function. File copy program, fprintf() and fscanf(). fread() and fwrite() function. Writing and reading records from binary file, Appending, modifying and deleting a record from file, Random access functions fseek(), rewind(), flushall(), remove(),rename()	10

Text Books:
1. Y.P.Kanetkar," Letus C ", BPBPublication
2. E.Balaburuswamy ," Programming in C",TataMacgrawHill
Reference Books:
1. Goterfried ,"Programming in C ",Shaums'Series
Website Resources: https://www.w3schools.com/c/ https://www.javatpoint.com/c-programming-language-tutorial https://www.geeksforgeeks.org/c-programming-language/

Semester II

Course code: CAS41MMP104 **Course name:** Practical based on Advanced C Programming **Course category:** Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.
Course Objectives: student get familiar with the basic concepts of data structures and algorithms, Get familiar with basic techniques of algorithms. Student get familiar with basic concepts about stacks, queues, lists, and hence student able to implement practically searching techniques.
Course Outcomes: At the end of the course, the students will be able to -
CO1: The course aims to provide exposure to problem-solving and principles through programming.
CO2: It aims to train the student to the basic concepts of the C programming language. This course involves a lab component which is designed to give the student hands-on experience with the concepts
CO3: Write the C code for a given algorithm.
CO4: Identify tasks in which the numerical techniques learned are applicable and apply them to write programs, and hence use computers effectively to solve the task

Contents

Unit	Content	Teaching hours
1	C Program to Check Prime or Armstrong Number Using User defined Function, C Program to Reverse a Sentence Using Recursion ,C programs based on Preprocessor directives, C program to calculate the power using recursion ,C programs based on Preprocessor directives.	04
2	C Program to Store Information of a Student Using Structure,C Program to Add Two Distances (ininch-feetsystem) using Structures,C Program to Calculate Difference Between Two Time Periods,Program to demonstrate Unions, Program to demonstrate pointers.	04
3	Program to demonstrate file handling ,C Program to Write a Sentence to a File,C Program to Read the First Line From a File ,C Program to Display its own Source Code as Output,C program to use various file handling functions.	04

Text Books:
3. Y.P.Kanetkar," Letus C ", BPBPublication
4. E.Balaburuswamy ," Programming in C",TataMacgrawHill
Reference Books:
2. Goterfried ,"Programming in C ",Shaums'Series
Website Resources: https://www.programiz.com/c-programming/examples https://www.javatpoint.com/c-programming-language-tutorial https://www.geeksforgeeks.org/c-programming-language/

Semester II

Course code: CAS41VSP102 **Course name:** Structured Query Language **Course category:** Vocational skill course
Credits: 2 **Teaching scheme:** L-0 P-4 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer Programming
Course Objectives: To understand the concept of Database management system
Course Outcomes: At the end of the course, the students will be able to -
CO1: Ability to implement and use database structures, including creation, Insertion, Deletion of table.
CO2: Understand of basics of database management.
CO3: Understand and analyze data for programming

Contents

Unit	Content	Teaching hours
1	Design and draw E-R diagrams,Study of 3-Levelarchitecture ,Study of MySQL,Download and Install Xampp server for MySQL,Creating a new user account and a database Creating scheme as for the database	04
2	Study of basic operations DDLcommands, Study of DML commands, Study of DCL commands ,Study of Constraints:Rule1&2, advanced constraints like primary key, foreign key, unique and check constraints on tables with examples,Write & execute queries using select command using where, groupby, orderby and having clauses.	04
3	Study of Basic Operations of Relational Algebra with examples (Union, Intersection, difference & Cartesian Product), Study of Selection and Projection Operations with examples, Study of Join(natural,Inner,outer,left&right) and Division Operations with examples ,Study of Single Row Functions with examples Study of conversion functions with examples.	04

Web site Resources:1.https://www.redhat.com/sysadmin/linux-skills-home-lab

Semester II

Course code: CAS41SEL102 **Course name:** Database Management System **Course category:** Skill Enhancement Course
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer Programming
Course Objectives: To understand the concept of Database management system
Course Outcomes: At the end of the course, the students will be able to -
CO1: Recognize and understand the basic concepts of database, knowledge, classifications of Architecture of database, Database users, define advantages of database.
CO2: Describe the components of data base system, define transaction, data modeling
CO3: Draw E-R diagram, schema diagram, classify attributes.

Contents

Unit	Content	Teaching hours
1	**Introduction to basic concepts of DBMS:** Database System Application ,Purpose of Database System ,DatabaseArchitecture:3-Level architecture ,Database Users & Administrators Responsibilities ,Functional Components of Database system: Storage& Query Processor Transaction Management	10
2	**Data Modeling &Design :**Type of Data Model: Relation Data Model ,E-R Data Model Object Based Data Model ,Semi-Structured Data Model ,Hierarchical & Network Data Model. **E-R Data Model:** Entity, Entity set, Entity types, Attributes, Types of Attributes, E-Rdiagram, Mapping Cardinalities, Data Association Constraints: Integrity constraints I&II Database Design: Overview of Design Process, Designing Phase, Normalization(1NF,2NF,3NF)	10
3	**Relational Data Model:**Basic Structure ,Database Schema ,Integrity Rules ,E.F.Codds Rules Relational Algebra: Union, Intersection, Difference, Cartesian Product, Selection, Projection, Join: Natural & Outer Join, Division,Trigger, Stored procedure with advantages and	10

Text Books:
1.Korth,Siberschat z,"Database System concepts".
Reference Books:
1.B.Desai,"An Introduction to Database System".

Semester: THIRD

Syllabus
Semester III

Course code: CAS41MML201 **Course Name**: Non Linear Data Structures **Course Category**: Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of computer programming and aware about data.
Course Objective:Course Objectives: Student get familiar with basic and advance concepts about Linear and Nonlinear data structure like, stacks, queues, linked lists, tree, graph, hashing etc. and hence student be able to implement practically searching and sorting techniques.
.
Course Outcomes: After completion of the course the student will be able to:
CO1: Remember the use of nonlinear data structures, including Tree, Graph, Hashing etc.
CO2: Understand of basic and some advance terminology of searching and sorting.
CO3: Understand and analyse the concepts of Hashing and Rehashing.
CO4: Apply and implement the linear and nonlinear data structure concepts.

Contents

Unit	Content	Teaching hours
1.	**Graph:** Introduction, Types of graph, Linked Representation of Graph. **Operations on Graph**-Traversing, Insertion and Deletions. Breadth First Search Algorithm Depth First Search Algorithm	06
2.	**Tree:** Introduction, Binary Tree, Representing Binary tree in memory. **Operations on Tree**:Traversing, Insertion and Deletions. AVL Tree, Binary Tree.	08
3.	**Searching and Sorting Techniques:** Searching Linear Search, Binary Search, Sorting, Bubble sort, Selection sort, Insertion sort, Shell sort, Merge Sort.	08
4.	**Hashing Techniques:**Hash Functions, Separate Chaining, Open Addressing, Rehashing – Extendible Hashing.	08

Text Book:

1. Fundamentals of Data structures, Horowitz & Sahani, Galgotia publication.

Reference Book:

1. Data Structures, Seymour Lipchitz, Tata McGraw Hill Education.
2. Data Structures, Tannenbaum, PHI publication.

Syllabus
Semester III

Course code: CAS41MML202 **Course name:** Object Oriented Programming C++ **Course category:** Major Mandator
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of computer Programming and aware about data.
Course Objectives: To introduce the concept of object oriented programming concepts using C++.
Course Outcomes: After completion of the course the student will be able to:
CO 1: Remember object oriented programming concepts using C++.
CO 2: Understand the use of control structures and arrays in object oriented terminology.
CO 3: Acquire knowledge of function prototyping, recursion, function overloading, operator overloading.
CO 4: Apply concepts like Classes and objects, control structures in C++, constructors, destructors, inheritance.

Contents

Unit	Content	Teaching hours
1	**Principal of Object Oriented Programming:** Object Oriented Programming Paradigm, Benefits of OOP, and Applications of OOP, What is C++? Structure of C++ Program, Creating Source File, Compiling and Linking. Variables, Identifiers and Constants, Data Types, Operators in C++	6
2	**Control Structures:** - Simple if, if...else statement, nested if, else if ladder, Relational operator, switch structure.**Arrays: -** Introduction to arrays in C++, Array declaration, Array initialization, Types of arrays in C++.	8
3	**Functions**–Introduction, Function Prototyping, Inline Function, Recursion, Function Overloading, Operator Overloading.**Class and Object**: Specifying Class, Defining Member Functions, Arrays with a Class, Memory Allocation for Objet, Static Data Member, Static Member Function, Objects as Function Arguments, Returning Objects.Working of queue	8
4	**Constructor & Destructor**- Parameterized Constructor, Multiple Constructor in a Class, Copy Constructor, Dynamic Constructor, and Destructor. **Inheritance**: Introduction to Inheritance, Defining Derived Classes,Single Inheritance, Multilevel Inheritance, Multiple Inheritance, Hierarchical Inheritance, Hybrid Inheritance	8

Text Books:

1. Object Oriented Programming with C++, E Balgurusamy, McGraw-Hill Education (India) Pvt. Limited.
2. C++ for Beginners ….. Masters, Ankiot Asthana, New age international Publisher.

Reference Book:

1. Object Oriented Programming in C++, Robert Lafore, Sams Publishings.
2. C++ The Complete Reference, Herbert Schildt 4th Edition.

Online Resources:

1. https://www.learncpp.com/cpp-tutorial/object-sizes-and-the-sizeof-operator/
2. https://www.w3schools.com/cpp/
3. https://www.programiz.com/cpp-programming
4. https://www.learn-cpp.org/
5. https://cplusplus.com/doc/tutorial/program_structure/

Syllabus
Semester III

Course code: CAS41M ML203 **Course name:** Fundamentals of Computer Networks **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: A foundational understanding of key concepts in computer science or information technology, covering basic programming concepts and fundamental computer architecture
Course Objectives: To learn fundamental networking concepts and terminology related to network hardware, software models, and topologies and compare key networking architectures and protocols including reference models like OSI and TCP/IP, understand the functions and services of networks and how they enable communication.
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: Knowledge of fundamental networking concepts including architectures, topologies, protocols, addressing schemes, and transmission media fundamentals.
CO 2: Understanding of key network reference models including OSI and TCP/IP from an architectural perspective along with understanding addressing & routing schemes.
CO 3: Understanding the basis behind data link layer concepts of framing, error control, flow control and other methods along with working of data link protocols.
CO 4: Understanding and implementing Switching and its modes.

Contents

Unit	Content	Teaching hours
1	**Introduction:** What are Computer Networks? Data communication & its components, **Data flow and its Types**: simplex, half-duplex, and full-duplex. **Networking terms and concepts:** IP addresses, nodes, routers, switches, ports, etc. **Uses of computer networks**, **Network hardware**: PAN, LAN, MAN, WAN, Wireless networks, Home networks, Internetworks, Network Topology & its Types	6
2	**Network software and models**: Protocol Hierarchies, Design Issues for the Layers, Connection-Oriented Versus Connectionless Service, Service Primitives, The Relationship of Services to Protocols. **Reference models:** the OSI Reference Model, The TCP/IP Reference Model, Comparison of OSI and TCP/IP Models**, Example networks:** The Internet, Third-Generation Mobile Phone Networks, Wireless LANs -802.11.	8
3	**Addressing in TCP/IP Protocol:** physical (link) addresses, logical (IP) addresses, port addresses, and specific addresses. **The physical layer:** guided transmission media, unguided transmission media (wireless), **Communication satellites:** Geostationary Satellites, Medium-Earth Orbit Satellites, Low-Earth Orbit Satellites **Digital Modulation:** Baseband Transmission, Passband Transmission	8
4	**Multiplexing:** Multiplexing, its need, and its types - Frequency-division	8

	Multiplexing (FDM), Wavelength Division Multiplexing (WDM), Time Division Multiplexing, **Switching and its modes:** Types of switching modes - Store-and-forward, Cut-through, Fragment-free. **The Data Link Layer:** Its functions and services provided to the network layer, Line Discipline, Framing, Error Control, Flow Control, Types of Data Link Protocols - Synchronous Data Link Protocol (SDLC), High-Level Data Link Protocol (HDLC), Serial Line Interface Protocol (SLIP), and others	

Text Book:

1. Computer Networks, Andrew S. Tanenbaum, Prentice Hall 5th Edition.

Reference Book:

1. Data Communications and Networking, Behrouz A. Forouzan, McGraw Hill 4th Edition.
2. A Top Down Approach, James F Kurose and Keith W Ross.

Syllabus
Semester III

Course code: CAS41MMP201 **Course name:** Practical based on Non Linear Data Structures **Course category:** Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.
Course Objectives: Practically, student get familiar with the basic concepts of data structures and algorithms, Get familiar with basic techniques of algorithms. Student get familiar with basic concepts about stacks, queues, Linked lists, Graph, Tree etc. and hence student able to implement practically searching techniques.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Implement and use linear and nonlinear data structures, including stacks, queues, linked lists.
CO 2: Understand of basic terminology data, data item, fields and data structures
CO 3: Understand and analyse the concepts of Searching and sorting
CO 4: Apply the concepts of linked list, Linked representation of Queue for specified applications.

Contents

Sr. No.	Title	Practical Hours
1.	Using previous knowledge of C or C++ to calculate the factor of any number.	02
2.	Perform matrix multiplication using programming skills of C or C++.	02
3.	Implement algorithm for Traversing of Graph in any programming language.	02
4.	Implementation of algorithm for insert new element in to Graph.	02
5.	Implement algorithm for delete element from the Graph.	02
6.	Write a Program in C to find Prime numbers between 1 to 2000 numbers.	02
7.	Make flowchart and Implement algorithm for Traversing of Tree.	02
8.	Draw neat diagram and Implement algorithm for Insert new element into Tree by algorithm.	02
9.	Perform algorithm for delete an element from the Tree by algorithm.	02
10.	Study and Implement algorithm for Binary Search by algorithm.	02
11.	Implementation of algorithm for Bubble sort in programming language.	02
12.	Implement algorithm for Selection sort in C or C++ language.	02

13.	Get practical analysis of Merge sort in any programming language.	02
14.	Write a Program in C for interchange the place of sentence 1 with sentence 2	02
15.	Project	02

Text Books:

1. Fundamentals of Data structures, Horowitz &Sahani, Galgotia publication.
2. Data Structures, Seymour Lipschutz, Tata McGraw Hill Education.

Reference Book:

1. Data Structures, Tannenbaum, PHI publication.
2. Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein.

Syllabus
Semester III

Course code: CAS41MMP202 **Course name:** Practical based on Object Oriented Programming C++ **Course category:** Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Knowledge of Computer components.
Course Objectives: To implement concepts of C++ practically and understand the object oriented programming concepts like object, class, constructor, overloading, and inheritance.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Gain understanding of the basic concepts of the C++ programming language.
CO 2: Understand about syntax of all the basic structures of C++ programming language.
CO 3: Learn to build programs based on various concept to solve real life problems
CO 4: Build nested structure of program to find solution to problem.

Contents

Sr. No.	Title	Practical Hours
1.	Program to Check Whether Number is Even or Odd using if else statement.	02
2.	Program to Generate Multiplication Table using for loop.	02
3.	Program to Check Whether a Number is Prime or not using control structures in C++.	02
4.	Program to Make a Simple Calculator to Add, Subtract, Multiply or Divide Using switch...case.	02
5.	Program to Calculate Power Using Recursion concept.	02
6.	Program to implement function overloading in order to compute power (m, n).	02
7.	Using the concept of function overloading Write function for calculating the area of triangle, circle and rectangle	02
8.	Programs to implement different types of inheritance : a. Simple inheritance b. Multiple inheritance Hierarchical inheritance	02
9.	Program for multiplication of two matrices using multidimensional arrays.	02
10.	Program to demonstrate constructor overloading	02
11.	Project	10

Text Books:

1. Object Oriented Programming with C++, E Balgurusamy, McGraw-Hill Education (India) Pvt. Limited
2. C++ for Beginners Masters, Ankiot Asthana, New age international Publisher

Reference Book:

1. Object Oriented Programming in C++, Robert Lafore, Sams Publishings
2. C++ The Complete Reference, Herbert Schildt 4th Edition.

Online Resources:

1. https://www.learncpp.com/cpp-tutorial/object-sizes-and-the-sizeof-operator/
2. https://www.w3schools.com/cpp/
3. https://www.programiz.com/cpp-programming
4. https://www.learn-cpp.org/
5. https://cplusplus.com/doc/tutorial/program_structure/

Syllabus
Semester III

Course code: CAS41VSP201 **Course name:** Advance Excel **Course category:** Vocational Skill Course
Credits: 2 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic of computer Programming and aware about data.
Course Objectives: Student get familiar with Advance Excel And its Features with goal seek, Advance Filter, Pivot table, VLOOKUP, HLOOKUP etc.
Course Outcomes: After completion of the course the student will be able to:
CO 1: Use advanced functions and productivity tools to assist in developing worksheets
CO 2: Manipulate data lists using Outline, Auto filter and Pivot Tables.
CO 3: Use consolidation to summaries and report results from multiple worksheets. Record repetitive tasks by creating Macros.

Contents

Sr. No.	Title	Practical Hours
1.	Practical based on Number Formatting	02
2.	Practical based on Data validation and Graph	
3.	Practical Based on Conditional formatting.	02
4.	Create a marksheet in Excel with following condition Heading – Dr. G. Y. Pathrikar College Subject – Any Five subject with Total, Percentage and Result Result condition – If Per >=80 ‘A’ Garde Per >=60 ‘ B’ Grade Per > 45 ‘C’ Grade Per>=35 ‘Pass’ Otherwise ‘Fail’	02
5.	Practical Based on Filter Or (Apply Filter to above table).	02
6.	Practical Based on Chart with formatting.	02
7.	Practical based on Formula.	02

8	Basic Formulas SUM, AVERAGE, COUNT, MAX, MEDIAN, MIN	02
9	Time Formulas: TODAY, NOW, DATEDIF, YEAR, MONTH, DAY	02
10	Logical Formulas: IF, OR, AND	02
11	Create a sheet and apply Trace Precedent, Trace Dependent on that sheet.	02
12	Practical based on VLOOKUP.	02
13	Practical based on HLOOKUP.	02
14	Practical based on Data validation	02
15	Crate a worksheet and apply goal seek, subtotal on it.	02
16	Practical based on Import data from web, word etc. in excel	02
17	Practical based on Pivot table.	02
18	Practical based on DatVisualization	02
19	Practical Based on macros.	02
20	Study of Dashboard	02
21	Create Dashboard in Excel	02
22	Project	20

Text Books:

1. Excel 2016 Bible, John Walkenbach

Reference Book:

1. Excel 2016 All-In-One For Dummies, Greg Harvey.
2. Excel 2016: Pivot Table Data Crunching Author: Bill Jelen, Michael Alexander.

Semester: FOURTH

Syllabus
Semester IV

Course code CAS41MML204 **Course name**: Advanced Database Management System **Course category**: Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic understanding of database management system.
Course Objectives: To understand the advanced concept of database management system.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Recognize and understand the working of query processor
CO 2: Describe database recovery mechanism.
CO 3: Gain a better understanding of database concurrency management.
CO 4: Understanding Concurrency control & Locking Scheme

Contents

Unit	Content	Teaching hours
1.	**Query Processing:** Introduction to query processing with an example, General Strategies for Query Processing, Transformation into an Equivalent Expression, Expected Size of Relations in the Response, Statistics in Estimation, Query Improvement	06
2.	**Query Evaluation**: Query Evaluation, Evaluation of Calculus Expressions, View Processing, A Typical Query Processor **Recovery& Recovery Schemes:** Reliability- Types of Failures, Types of Errors in Database Systems, Possible Detection Schemes, Audit Trails, Recovery Schemes, Transactions- States of a Transaction, Properties of a Transaction	08
3.	**Other Recovery Schemes & Cost Comparison:** Recovery in a Centralized DBMS, Reflecting Updates to the Database and Recovery, Buffer Management, Virtual Memory, and Recovery, Other Logging Schemes, Cost Comparison **Concurrency Management:** Introduction to concurrency management, Serializability& its algorithm, Concurrency Control	08
4.	**Concurrency control & Locking Scheme:** Locking Scheme- two phase locking, DAG database storage structure, Timestamp-Based Order, Deadlock Detection and Recovery, Deadlock Avoidance, Database Security Policies, Database Authorization, Identification and Authentication, Distributed Systems & Cryptography and Encryption	08

Text Book:

1. An Introduction to Database Systems, Bipin C.DESAI

Reference Book :

1. Fundamentals of Database Management, Elmasri, Navate, 6th Edition
2. Database Management System, Raghu Ramakrishnan, 2nd Edition.

Syllabus
Semester IV

Course code: CAS41MML205	**Course name:** Web Technologies	**Course category:** Major Mandatory
Credits: 2	**Teaching scheme:** L-2 P-0	**Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of internet and basic concept of web site		
Course Objectives: To understand how CSS can enhance the design of a webpage. And To understand the XML to represent the web data		
Course Outcomes: After completion of the course the student will be able to:		
CO 1: Independently understand the difference between HTML and XHTML		
CO 2: Develop different parts of a web page using Style Sheets		
CO 3: Understand the difference between Client Side Programming and Server Side Programming.		
CO 4: Design with the JavaScript language		

Contents

Unit	Content	Teaching hours
1.	**HTML**: Introduction, web server, web client/ browser, HTML tags, **C**ommonly used HTML commands, structure of HTML program, formatting, text styles, text effects, HTML lists, types of lists, adding graphics to HTML document, Creating tables, linking documents, frames. **HTML Forms:** Form, Form Attributes, HTML Form Controls, Field set and Legend, Tab Index Attribute, The Form Object's Methods	06
2.	**Cascading Style Sheets:** Introduction, Features, Core Syntax, How to add CSS, Inline CSS, External CSS, Internal CSS, Properties of CSS, Class. ID **CSS Advanced Effects:** CSS Border Images, backgrounds, color, color keywords, Gradients, Shadows, Text effects, Web fonts**.**	08
3.	**Client- Side Programming:** Introduction to JavaScript, JavaScript in Web Pages, JavaScript-Advantages, Features and Limitations, Writing JavaScript into HTML, Datatypes and Literal, Variables, Operators and Expressions in JavaScript. **JavaScript Programming Constructs:** Conditional Statements, Loops, Loop Controls, Functions in JavaScript, User defined functions, Placing Text in a Browser, Dialog Boxes.	08

4.	**Introduction to XML:** XML Introduction, features, applications, HTML vs XML, Syntax, Declarations, Tags and Elements, XML Attributes, Structure of XML, XML Entities, Entity types, XML Comments, XML Tree.	08

Text Books:
1. Web Enabled Commercial Application Development Using HTML, JAVASCRIPT, DHTML and PHP, Ivan Bayross, BPB PUBLICATION, Fourth Edition.
2. Beginning XML, 4th Edition, David Hunter, Jeff Rafter, Joe Fawcett, Eric van der Vlist, Danny Ayers, Jon Duckett, Andrew Watt, Linda McKinnon
Reference Books:
1. HTML & CSS: The Complete Reference, Thomas A. Powell, Tata McGraw Hill, Fifth Edition
2. WEB PROGRAMMING with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, Jones and Bartlett Publishers, Inc., First Edition

Syllabus
Semester IV

Course code: CAS41MML206 **Course name:** Advanced Computer Networks **Cours category:** Major Mandatory	
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20	
Pre-requisites: A foundational understanding of key concepts in computer networks like network models, types of transmission media, topology, addressing, etc.	
Course Objectives: To learn advanced networking concepts and terminology related to computer networks and the functions of the layers of networking models. And To gain a basic understanding of information security over networks.	
Course Outcomes: After completion of the course the student will be able to demonstrate:	
CO 1: Comprehensive Understanding of Networking Protocols and Technologies	
CO 2: Understanding of the Advanced Network Layer Concepts and Internetworking	
CO 3: Advanced Understanding of Transport and Application Layer Protocols	
CO 4: To study Network Simulation & Simulation tool	

Contents

Unit	Content	Teaching hours
1.	**The Medium Access Control Sublayer – Multiple Access Protocols:** ALOHA, Carrier Sense Multiple Access Protocols, Collision-Free Protocols, Wireless LAN Protocols. **Ethernet**: Classic Ethernet Physical Layer, Classic Ethernet MAC Sublayer Protocol, Switched Ethernet, Fast Ethernet.	06
2.	**Wireless LANs:** The 802.11 Architecture and Protocol Stack, **Data Link Layer Switching**: Uses of Bridges, Learning Bridges, Spanning Tree Bridges, Repeaters, Hubs, Bridges, Switches, Routers, and Gateways, Virtual LANs. **THE NETWORK LAYER:** Store-and-Forward Packet Switching, Services Provided to the Transport Layer, Wireless networks – Cellphones, 5G.	08

3.	**Routing Algorithms**: The Optimality Principle, Shortest Path Algorithm, Flooding, Distance Vector Routing, Link State Routing, **Internetworking:** How Networks Differ, How Networks Can Be Connected. **The Network Layer in The Internet:** The IP Version 4 Protocol, IP Addresses, IP Version 6, **THE TRANSPORT LAYER**: The Transport Service, Services Provided to the Upper Layers, Transport Service Primitives.	08
4.	**Elements of Transport Protocols:** Addressing, Connection Establishment, Connection Release, Error Control and Flow Control, **THE APPLICATION LAYER: DNS—The Domain Name System**: The DNS Name Space, Domain Resource Records, Name Servers, **Electronic Mail:** Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery. Introduction to Network Simulation & Simulation tool - Cisco Packet Tracer.	08

Text Books:
1.Computer Networks, Andrew S. Tanenbaum, Prentice Hall 5th Edition
2.Computer Networking A Top-Down Approach, James F. Kurose, Keith W. Ross, Pearson Education, Inc. 7th Edition.
3.Data Communications and Networking, Behrouz A. Forouzan, McGraw Hill 4th Edition
Reference Books:
1. Networking Essentials Companion Guide, Cisco Networking Academy, CISCO Press.
2. Computer Networks and Internet Protocol, Prof. Soumya Kanti Ghosh, Prof. Sandip Chakrabort, IIT, Kharagpur
Online Resources: 1. https://onlinecourses.nptel.ac.in/noc24_cs19/preview 2. https://www.netacad.com/courses/packet-tracer

Syllabus
Semester IV

Course code: CAS41MMP203 **Course name**: Practical based on Advanced Database Management System **Course category**: Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic understanding of database management system and RDBMS.
Course Objectives: To understand the advanced concepts of DBMS.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Understand and analyse the working of query processor
CO 2: Implement various methods for SELECTION and JOIN.
CO 3: Understand and analyse PROJECT, SET, AGGREGATE operation. Learn database authorization and authentication, and encryption methods.

Contents

Sr. No	Title	Practical Hours
1	Learn and analyse the steps of query processing in database management system.	02
2	Stepwise study of query optimization technique in database management.	02
3	Translating SQL queries into relational algebra through its algorithm.	02
4	Demonstrate the use of algorithm for external sorting of the data in DBMS.	02
5	Learn data searching method with the use of simple selection command.	02
6	Illustrate the effect of search method through the implement of complex selection	02
7	Show the implement disjunctive selection condition and analyse its effect on data	02
8	Demonstrate the use of algorithm for JOIN operation and implement the same.	02
9	Show the proper use algorithm for PROJECT and SET command in database management.	02
10	Implement aggregate operation on the database and evaluate the same.	02
11	Show the effect of outer JOIN through its query implementation.	02
12	Learn database authorization and authentication methods.	02
13	Learn database authorization and authentication methods.	02
14	Describe the use of cryptography and encryption method.	02
15	Project	04

Text Books: 1. Fundamentals of Database Management, Elmasri, Navate 6th Edition.
References Books: 1. An Introduction to Database Systems, Bipin C.DESAI
2. Database Management System, Raghu Ramakrishnan, 2nd Edition.
Online References: 1. NPTEL/ SWAYAM online courses 2. https://www.geeksforgeeks.org/dbms/ 3. https://www.javatpoint.com/dbms-tutorialAn Introduction to Database Systems, Bipin C.DESAI 3. Database Management System, Raghu Ramakrishnan, 2nd Edition.

Syllabus
Semester IV

Course code: CAS41MMP204 **Course name:** Practical based on Web Technologies **Course category**: Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Course Objectives: Student able to implement various Web sites using HTML tags, CSS, JavaScript, and able to implement Servlet programs.
Course Outcomes: After completion of the course the student will be able to:
CO 1: To understand about syntax of all the basic structures of HTML and CSS
CO 2: To understand the programming concept of JavaScript and hence able to create web page
CO 3: To create web pages (websites) or applications that run in a web browser
CO 4: To enhance and build the web data through various web techniques.

Contents

Sr. No	Title	Practical Hours
1	Programs to demonstrate the basic HTML tags with Text styling, lists and tables.	02
2	Create a web page using HTML which demonstrates the Relative URL, Absolute URL and HTML Frames.	02
3	Build a web page to demonstrate the HTML Form.	02
4	Create a web page using CSS.	02
5	Program to demonstrate Java script programming using variables, various operators.	02
6	Program to demonstrate Java script programming conditional statement.	02
7	Design a web page which demonstrate the use of Java Script Event and Java Script Object.	02
8	Implement the program which demonstrates the functions in JavaScript.	02
9	Create a web page which demonstrates the XML Attributes.	02
10	Program which broadcast the concept of XML Entities.	02
11	Create a model which demonstrates the various concept of Web Technology.	02

Text Books:
1.Jeffrey C.Jackson, “Web Technologies–A Computer Science Perspective”, Pearson Education, 2006
2.Robert. W. Sebesta, “Programming the World Wide Web”, Fourth Edition, Pearson Education, 2007.
Reference Books:
1.Web Enabled Commercial Application Development Using HTML, JAVASCRIPT, DHTML and PHP, Ivan Bayross, BPB PUBLICATION, Fourth Revised Edition.
2.WEB PROGRAMMING with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, First Edition

Syllabus
Semester IV

Course code: CAS41SEP201	**Course name**: PHP	**Course category**: Skill Enhancement Course
Credits: 2	**Teaching scheme:** L-0 P-4	**Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic understanding of PHP programming.		
Course Objectives: To understand the concept of Server side scripting language through PHP		
Course Outcomes: At the end of the course, the students will be able to –		
CO 1: Recognize and understand the use of variables, constants, arithmetic operators and logical operators in PHP.		
CO 2: Understand the concept of comparison operators, conditional statements, types of loop statements and types of array in PHP.		
CO 3: Develop a better understanding of various arrays, functions manipulating arrays and learn to create user defined functions in PHP.		
CO 4: Understand various string manipulating functions and learn file handling, session and cookie management in PHP.		

Contents

Sr. No	Title	Practical Hours
1	Learn the stepwise process of downloading and installing PHP on Windows.	02
2	Learn commenting & escape sequencing in PHP through a script writing.	02
3	Study declaration and usage variables and constants in PHP script.	02
4	Implementation of arithmetic operators through PHP script.	02
5	Demonstrate the use of logical operators in server side scripting.	02
6	Learn comparison operators such as ==, >=, <=, != with the help of writing a script.	02
7	Study conditional statements such as If-then, If-else, nested If-else, switch(), break, continue in PHP.	02
8	Loop statements like while(), do-while(), for() in PHP.	02
9	Demonstrate the use indexed array through a script in PHP.	02
10	Usage of associative array in PHP through a program implementation.	02
11	Manipulation of array with print_r(), count() functions in PHP.	02

12	Sorting of array with sort(), rsort(), assort(), arsort() functions in PHP.	02
13	Using array function such as array_push(), array_pop(), array_shift(), array_unshift(), array_combine() in PHP.	02
14	Effective use foreach() looping with array in PHP.	02
15	Declaration and definition of a user defined function for finding square root of a given number in PHP.	02
16	Implementing string operation through functions such as strlen(), strops(), strstr(), substr(), str_replace(), strtolower(), strtoupper() in PHP.	02
17	Demonstrate file reading operation with the use of fread() function in PHP.	02
18	Illustrate file writing operation with the use of fwrite() function in PHP.	02
19	Importance of Session and Session management using Session variables.	02
20	Use of Cookies in PHP through a program implementation.	02
21	Project	20

Text Books:

1. How to Do Everything with PHP & MySQL, VikramVaswani
2. Beginning PHP and MySQL, W Jason Gilmore

Reference Books:

1. .Learning PHP and MySQL, Michele E. Davis
2. Headfirst PHP & MySQL, Lynn Beighley, O′Reilly 1st edition

Online References:

1. NPTEL/ SWAYAM online courses
2. https://www.w3schools.com/php
3. https://www.javatpoint.com/php-tutorial

Semester: FIFTH

Syllabus
Semester V

Course code: CAS41MML301 **Course name:** Software Project Management **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-1 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: 1. Familiarity with programming languages and software tools 2. Ability to communicate effectively and manage interpersonal relationships in the teams.
Course Objectives: 1. Basic understanding of project management concepts like planning, scheduling, and resource allocation 2. Ability to identify, analyze, and propose solutions to given challenges.
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: To plan and manage project at each stage of the surface development life cycle
CO 2: To develop skills to manage people and project that various faces
CO 3: To learn about the activity planning and risk management using organization principles
CO 4: To deliver successful project and support the organization with strategic implementation

Contents

Unit	Content	Teaching hours
1	**Introduction to Software Project Management :** Introduction to Project Management Concepts, Role of Project Manager, Management Level, structure and its process ,Overview of Software Project Management (SPM), and Importance Project Spectrum,Software Development Life Cycle (SDLC),Tools and Techniques in Traditional Project Management	06
2	**Project Planning and Scheduling:**Project Scope Management:Defining Scope, Resources Work Breakdown Structure,process and importance,**Estimation Techniques:** Effort and Cost Estimation, COCOMO model **Project Scheduling:** Gantt Charts, Project Evaluation & Review Technique (**PERT**), Critical Path Method(**CPM**),**Project Evaluation Technique:** Earned Value Analysis(**EVA)**, Project Metrics, Key Performance Indicator(**KPI**),**Risk Management** in Software Projects, Categories of risk, **Risk Management process:** Identification, Potential risk, Prioritization risk	08
3	Software Requirement Specification(**SRS**) ,Software Configuration Management(**SCM**)**Software Re-Engineering:** Re - development, Re – engineering ,**Software Maintenance:** Problem Identification and resolve the problems ,Software Quality Assurance (**SQA**)	08

	Data Analytics in Project Management	
4	**Introduction to Unified Modeling Language :**Overview of Unified Modeling Language,Types of Diagram and models ,Use Case Model, Sequence Model,Activity Model,Class Model Object Model.	08

Text Books:
1. Software Project Management - Bob Hughes, Mike Cotterell, and Rajib Mall, TMH Publications
2. Agile Project Management For Dummies- Paperback, Layton Mark C.,John Wiley & Sons Inc
Reference Books:
1. Project Management and Tools & Technologies – An overview, Shailesh Mehta,Shroff Publishers & Distributors Pvt Ltd
2. Software Project Management - Walker Royce, Pearson
Online References: 4. NPTEL/ SWAYAM courses 5. https://www.w3schools.com/ 6. https://www.geeksforgeeks.org/software-engineering-software-project-management-spm/ 7. https://www.tutorialspoint.com/software_engineering/software_project_management.htm

Syllabus
Semester V

Course code: CAS41M MP301 **Course name:** Practical Based on Software Project Management
Course category: Major Mandatory

Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20

Pre-requisites: Basic of computer application.

Course Objectives: To make students understand about the practical implementation of C programs

Course Outcomes: Understanding Project Management Framework, Perform Cost & Time Estimation and Identify risk, Implementation of the project

CO1: Ability to Plan and Organize a Software Project
CO2: Proficiency in Project Scheduling, Budgeting, and Risk Management
CO3: Implementation of Agile and Quality Management Practices

Text Books:
1. Software Project Management - Bob Hughes, Mike Cotterell, and Rajib Mall, TMH Publications
2. Agile Project Management For Dummies- Paperback, Layton Mark C.,John Wiley & Sons Inc
3. Learning UML 2.0 - Russ Miles & Kim Hamilton- O'Reilly

Reference Books:
1. Project Management and Tools & Technologies – An overview,Shailesh Mehta,Shroff Publishers & Distributors Pvt Ltd
2. Software Project Management - Walker Royce, Pearson
3. The Unified Modeling Language Reference Manual -James Rumbaug, Ivar Jacobson, Grady Booch- Addison Wesley

Online Resources:
1. NPTEL/SWAYAM courses
2. https://www.w3schools.com/
3. https://www.geeksforgeeks.org/software-engineering-software-project-management-spm/
4. https://www.tutorialspoint.com/software_engineering/software_project_management.htm

Sr. No.	Steps to be Observations & Analysis to done using Case studies (as per course coordinator)	Practical Hours
1	**Project Charter Creation** Define a project vision, objectives, stakeholders, scope, and constraints. Create a charter document outlining roles and responsibilities.	02
2	**Work Breakdown Structure (WBS) Development** Break the project into smaller, manageable tasks. Analysis which tools to visualize tasks in graphical .	02
3	**Project Scheduling** Develop a timeline using Microsoft TOOLS. Identify task dependencies and milestones.	02
4	**Risk Management Plan** Identify potential risks in a software project. Categorize risks (technical, financial, operational) and create mitigation strategies.	02
5	**Project Planning and Execution** Simulate a user story and Find KPI's while grooming. **Conduct daily stand-up meetings on metrics for retrospectives.**	02
6	**Cost Estimation and Budgeting** Prepare a budget considering development, testing, and deployment costs.	02
7	**Change Management Exercise** Simulate a scenario where requirements change mid-project.	02
8	**Quality Assurance and Testing Strategy** Define test cases, acceptance criteria, and review defect tracking.	02

9	**Stakeholder Reporting SRS format** Create a project status report or dashboard. Conduct a mock stakeholder meeting to present project progress.	02
10	**Post-Project Review and Lessons Learned** Document challenges, best practices, and improvements for future projects.	02
11	**Project**	10

Syllabus
Semester V

Course code: CAS41MML302 **Course name: Core Java** **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Knowledge of Programming language C or C++ is helpful but not mandatory.
Course Objectives: Learn OOP concepts helps to become Java developer.
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.
CO 2: Able to solve real world problems using OOP techniques (Objects and Classes).
CO 3: Able to understand the use of Inheritance and Interface.
CO 4: Apply classes, objects, members of a class and the relationships among them needed for a specific problem.

Contents

Unit	Content	Teaching hours
1.	**An Introduction to Java:-** A Short History of Java o Features or buzzwords of Java o Comparison of Java and C++ , Java Environment o Simple java program , Java Tools – jdb, javap, javadoc, Java IDE – Eclipse/Net Beans , Types of Comments , Data Types, Final Variable, Declaring 1D, 2D array.	06

2.	**Objects and Classes:-** Defining Your Own Classes, Access Specifies (public, protected, private, default), Array of Objects , Constructor, Overloading Constructors and use of 'this' Keyword , static block, static Fields and methods, Predefined class – Object class methods (equals(), toString(), hashcode(), getClass()), Inner class, Creating, Accessing and using Packages, Wrapper Classes.	08
3.	**Inheritance and Interface:-** Inheritance Basics (extends Keyword) and Types of Inheritance, Super class, Subclass and use of Super Keyword , Method Overriding and runtime polymorphism, Use of final keyword related to method and class, Use of abstract class and abstract methods, Defining and Implementing Interfaces.	08
4.	**Exception Handling:-** Dealing Errors, Exception class, Checked and Unchecked exception, Catching exception and exception handling, Creating user defined exception Strings, Streams and Files o String class and StringBuffer Class, Formatting string data using format() method.	08

Text Books:
1Complete reference Java, Herbert Schildt, 5th edition, McGraw-HIll
2.Beginning Programming with Java For Dummies (5th Edition)
3.Head First Java: A Brain-Friendly Guide (2nd Edition)
Reference Books:
1.Programming with Java, A primer, E. Balagurusamy, 4th edition
2..ssJava: Programming Basics for Absolute Beginners (1st Edition)
Online Resources: 1. https://www.tutorialspoint.com/java/index.htm 2. https://www.javatpoint.com/java-tutorial

Syllabus
Semester V

Course code: CAS41M MP302 **Course name**: Practical Based on Core Java **Course category**: Major Mandatory
Credits: 01 **Teaching scheme:** L-0 P-1 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Knowledge of Programming Language C or C++ is helpful
Course Objectives: 1. Learn how to implement object-oriented designs with Java. 2. Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.
CO 2: Validate input in a Java program.
CO 3: Able to understand the use of abstract classes.
CO4: classes, objects, members of a class and the relationships among them needed for a specific problem.

Contents

Sr. No	Title	Practical Hours
1	Broadcast a program to print ―Hello World‖ on the screen.	02
2	Implement the Java program to display the various * pattern.	02
3	Broadcast the Java program to print the area and perimeter of a circle	02
4	Create a Java program to add two binary numbers	02
5	Build a Java program to reverse a string.	02
6	Program to count the letters, spaces, numbers and other characters	02

7	Implement a Java function that calculates the sum of digits for a given char array	02
8	Find the smallest and largest element from the array.	02
9	Designed a class that demonstrates the use of constructor and destructor.	02
10	Program to implement single level inheritance.	02
11	Project	10

Text Books:

1.Complete reference Java, Herbert Schildt,5th edition, McGraw-HIll

2. Programming with Java, A primer, E. Balagurusamy, 4th edition

References Books:sss

1.Beginning Programming with Java For Dummies (5th Edition)

2.Java: Programming Basics for Absolute Beginners (1st Edition)

Online References:

1. NPTEL/ SWAYAM online courses
2. https://www.geeksforgeeks.org/dbms/
3. https://www.javatpoint.com/dbms-tutorial

Syllabus
Semester V

Course code: CAS41MML303 **Course name:** Data Science **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Good mathematical background and programming skills sufficient enough to learn new languages and software are required. Basic knowledge of statistics.
Course Objectives: Conceptual nature of data science
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: To develop fundamental knowledge of concepts underlying data science
CO 2: To develop practical data analysis skills, which can be applied to practical problems
CO 3: To explain how math and information sciences can contribute to building better algorithms and software
CO 4: To develop applied experience with data science software, programming, applications and processes

Contents

Unit	Content	Teaching hours

1.	**Introduction to Data Science Concepts:**Causality and Experiments, Data Preprocessing: Data cleaning, Data reduction, Data transformation, Data discretization.**Visualization and Graphing:**Visualizing Categorical Distributions, Visualizing Numerical Distributions, Overlaid Graphs, plots, and summary statistics of exploratory data analysis, Randomness, Probability, Introduction to Statistics,Sampling, Sample Means and Sample Sizes.	06
2.	**Data Science Technology Stack:**Rapid Information Factory Ecosystem, Data Science Storage Tools, Data Lake, Data Vault, Data Warehouse Bus Matrix, Data Science Processing Tools ,Spark, Mesos, Akka , Cassandra, Kafka, Elastic Search, R ,Scala, Python, MQTT, The Future Layered Framework: Definition of Data Science Framework, CrossIndustry Standard Process for Data Mining (CRISP-DM), Homogeneous Ontology for Recursive Uniform Schema, The Top Layers of a Layered Framework, Layered Framework for High- Level Data Science and Engineering Business Layer: Business Layer, Engineering a Practical Business Layer Utility	08
3.	**Three Management Layers**:Operational Management Layer, Processing-Stream Definition and Management, Audit, Balance, and Control Layer, Balance, Control, Yoke Solution, Cause-and-Effect, Analysis System, Functional Layer, Data Science Process **Retrieve Superstep :**Data Lakes, Data Swamps, Training the Trainer Model, Understanding the Business Dynamics of the Data Lake, Actionable Business Knowledge from Data Lakes, Engineering a Practical Retrieve Superstep, Connecting to Other Data Sources	08
4.	**Assess Superstep:**Assess Superstep, Errors, Analysis of Data, Practical Actions, Engineering a Practical Assess Superstep, 12 8 IV Process Superstep : Data Vault, Time- Person-Object-Location-Event Data Vault, Data Science Process, Data Science, Transform Superstep: Transform Superstep, Building a Data Warehouse, Transforming with Data Science, Hypothesis Testing, Overfitting and Underfitting, Precision-Recall, Cross-Validation Test **.Organize and Report Supersteps :** Organize Superstep, Report Superstep, Graphics, Pictures, Showing the Difference	08

Text Books:

1. Computational and Inferential Thinking: The Foundations of Data Science: AdiAdhikari and John DeNero e, e-book
2. Practical Data Science: Andreas François Vermeulen, APress
3. Principles of Data Science: SinanOzdemir, PACKT

Reference Books:

1. Data Science from Scratch: Joel Grus, O'Reilly
2. first Principle in python: Joel Grus, Shroff Publishers

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc24_cs19/preview

Syllabus
Semester V

Course code: CAS41MEL301 **Course name:** Multidimensional Computer Graphics **Course category:** Major Elective
Credits:3 **Teaching scheme:**L-03 P-01 **Evaluation scheme:**CA–60,ESE–40
Pre-requisites: Mathematical background and programming skills.
Course Objectives: Understand and learn the Geometrical Transformations in 2-Dimensional and 3-Dimensional perspectives.
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: Understand the basics of computer graphics, different graphics systems and applications of computer graphics.
CO 2: Use of geometric transformations on graphics objects.
CO 3: Analyze and apply clipping algorithms.
CO 4: Understand the basics of general animation functions.

Contents

Unit	Content	Teaching hours
1.	**Overview of Computer Graphics:**Introduction of Computer Graphics, Advantages of Computer Graphics, Application of computer graphics. Graphics Systems: Video Display Devices-Cathode Ray Tubes, Random Scan Systems, Raster Scan Systems, Color CRT Monitors, Liquid Crystal Monitors, Input Devices.	10
2.	**Output Primitives:** Output Primitives: Points and Lines, Line Drawing Algorithms, DDA Algorithm, Bresenham's Line Algorithm, Circle Generating Algorithms, Midpoint Circle Algorithm, Midpoint Ellipse Algorithm, Antialiasing of Lines, Method of Antialiasing, Increasing Resolution.	10
3.	Two Dimensional Transformations: Translation, Rotation, Scaling, Homogenous Coordinates for Translation, Homogenous Coordinates for Rotation, Homogenous Coordinates for Scaling, Composition of 2D Transformations, 2D Clipping-Point Clipping, Line Clipping, Sutherland and Cohen Subdivision Line Clipping Algorithm.	10

4.	Three-Dimensional Viewing, Projections and Clipping: Three Dimensional Viewing, Viewing Parameters, Transformation from World Coordinate to Viewing Coordinates, Projections-Parallel Projection, Perspective Projection, Types of Parallel Projection, Types of Perspective Projection.	10
5.	Introduction to Animation: Design of Animation Sequences, General Computer-Animation Functions, Raster Animations.	05

Text Books:

1. D. Hearn and M. Pauline Baker, Computer Graphics (C Version), Pearson Education, 2nd Edition, 2004.
2. Computer Graphics : A.P.Godse, (IIIrd Edition) ,Technical Publication

Reference Books:

1. Samit Bhattacharya. (2015). Computer Graphics. Oxford University Press.
2. Procedural Elements for Computer Graphics: D.F.Rogers

Online Resources:

1. NPTEL / SWAYAM lectures.

Syllabus
Semester V

Course code: CAS41M EP301 **Course name**: Practical Based on Multidimensional Computer Graphics **Course category**: Major Elective
Credits: 01 **Teaching scheme:** L-0 P-01 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Mathematical background and programming skills.
Course Objectives: Understand and learn the Geometrical Transformations in 2-Dimensional and 3-Dimensional perspectives.
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: Understand the basics of computer graphics, different graphics systems and applications of computer graphics.
CO 2: Use of geometric transformations on graphics objects.
CO 3: Analyze and apply clipping algorithms.
CO 4: Understand the basics of general animation functions.

Contents

Sr. No	Title	Practical Hours
1	Write a Program to draw a line.	02
2	Implement Program to draw a random line.	02
3	Write a Program to display text randomly.	02
4	Implement Program to draw a triangle.	02
5	Write a Program for DDA line drawing algorithm.	02
6	Demonstrate a Program to draw a Circle.	02
7	Implement Program to a draw Circle using the DDA circle drawing algorithm.	02
8	Write a Program for Bresenham's line drawing algorithm.	02
9	Implement Program to a draw Circle using the Bresenham's circle drawing algorithm.	02
10	Implement Program for 2D transformation: rotation of line.	02
11	Write a Program to implement Cohen Sutherland line Clipping algorithm.	02

12	Write a Program for 3D transformation	02
13	Write a Program for Animation.	02
14	Implement a Program to draw animation using increasing circles filled with different colors and patterns.	02
15	Write a Program to draw moving car	02

Text Books:

1. D. Hearn and M. Pauline Baker, Computer Graphics (C Version), Pearson Education, 2nd Edition, 2004.
2. Computer Graphics : A.P.Godse, (IIIrd Edition) ,Technical Publication

Reference Books:

1. Samit Bhattacharya. (2015). Computer Graphics. Oxford University Press.
2. Procedural Elements for Computer Graphics: D.F.Rogers

Online Resources:

1. NPTEL / SWAYAM lectures.

Syllabus
Semester V

Course code: CAS41MEL302 **Course name:** Advance PHP for Content Management System **Course category:** Major Elective
Credits: 03 **Teaching scheme:** L-04 P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic understanding of PHP, HTML and MySQL
Course Objectives: Understand and learn content management system, manage users, sessions, caches, handlers, access control, extensions, menus, languages, presentation, web hosting, error handling, SEF &RESTful services and real content with PHP.
Course Outcomes: After completion of the course the student will be able to :
CO 1: Understand of content management system and PHP
CO 2: Learn managing users, sessions, caches, handlers and access control
CO 3: Able to deal with Extensions, Menus, Languages and Presentation
CO 4: Understand Web hosting, SEF &RESTful services, Error handling and Real content

Contents

Unit	Content	Teaching hours
1.	**Introducing Content Management System and PHP:**CMS Architecture: Idea of Content Management System, technology for building CMS, CMS environment.Organizing Code: problem , discussion and consideration,exploring PHP and object design, framework solution Database and data objects: Problem , discussion and consideration, exploring PHP indirect referencing, framework solution	06
2.	**Managing Users, Sessions, Caches, Handlers and Access control:** Administrators, Users and Guests: Problem , discussion and consideration, exploring PHP array and SQL, framework solution. Sessions and Users: Problem , discussion and consideration, session data and scalability, exploring PHP framework of classes, framework solutions, creating a session, keeping session data tidy.Caches and Handlers: Problem , discussion and consideration, exploring PHP static elements and helpers, framework solutions.Access Control: Problem , discussion and consideration, exploring SQL, MySQL and PHP, framework solutions.	08

3.	**Dealing with Extensions, Menus, Languages and Presentation:** Handling Extensions: Problem , discussion and consideration, exploring PHP XML handling, framework solutions.Menus: Problem , discussion and consideration, exploring PHP array function, framework solutions. Languages: Problem , discussion and consideration, exploring PHP character set, framework solutions.Presentation Services: Problem , discussion and consideration, exploring PHP clarity and succinctness, framework solutions.	08
4.	**Learning Web hosting, SEF & RESTful services, Error handling and Real content:** Other Services: Problem, discussion and considerations, exploring PHP –files issues in web hosting, basic file and directory permissions, hosting and ownership, living with split ownership, avoiding split ownership.SEF and RESTful services: Problem, discussion, exploring PHP-PHP and HTTP, framework solution.Error Handling: Problem, discussion, exploring PHP-error handling, framework solution.	08

Text Books:

1. PHP5 CMS Framework Development- Martin Brampton. Second Edition, PACKT Publishers.
2. Build Your Own Database Driven Website Using PHP &MySQ -, Kevin Yank, Second Edition SitePoint Pty. Ltd.

ssReference Books:

1. Head First PHP & MySQL- Lynn Beighley and Laura Thomson
2. PHP: The Complete Reference - Herbert Schildt

Online Resources:

1. NPTEL/SWAYAM courses
2. https://www.w3schools.com/
3. https://www.freecodecamp.org/

Syllabus
Semester V

Course code: CAS41M EP302 **Course name**: Practical Based on Advance PHP for Content Management System **Course category**: Major Elective
Credits: 01 **Teaching scheme:** L-0 P-01 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic understanding of PHP, HTML and MySQL
Course Objectives: Understand and implement content management using PHP
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Create basic blog with post editing and viewing, user authentication, add photo gallery, albums.
CO 2: Implement basic image resizing, watermarking, users to edit text, images and contact information.
CO 3: Create CMS for a small business, basic contact form, user registration & authentication and building a news portal.

Contents

Sr. No	Title	Practical Hours
1	Create a basic blog with features to add, edit, and delete posts.	02
2	Implement basic user authentication (login/logout).	02
3	Display posts in a list view and provide a single-post view.	02
4	Build a simple photo gallery to upload, organize, and display images.	02
5	Allow users to create albums and categorize photos.	02
6	Implement basic image resizing and watermarking features.	02
7	Create a CMS for a small business to manage basic information (about us, contact, services).	02
8	Allow users to edit text, images, and contact information.	02
9	Implement a basic contact form, and Implement user registration and authentication.	02
10	Build a basic news portal with sections for different categories (e.g., politics, sports, technology).	02
11	Allow users to add, edit, and publish news articles. and Display news articles in a categorized list.	02

12	Project	08

Text Books:

1. Build Your Own Database Driven Website Using PHP &MySQ -, Kevin Yank, Second Edition SitePoint Pty. Ltd.
2. Head First PHP & MySQL- Lynn Beighley and Laura Thomson

References Books:

1. PHP5 CMS Framework Development- Martin Brampton. Second Edition, PACKT Publishers
2. PHP: The Complete Reference - Herbert Schildt

Online References:

1. NPTEL/SWAYAM courses
2. https://www.w3schools.com/
3. https://www.freecodecamp.org/

Syllabus
Semester V

Course code: CAS41VSP301 **Course name**: Android Application Development **Course category**: Vocational Skill Course
Credits: 2 **Teaching scheme:** L-0 P-4 **Evaluation scheme:** CA-30 ESE-20
Pre-requisites: Familiarity with basic programming concepts such as variables, loops, and functions, basic knowledge of Java or any object-oriented programming language, understanding of basic file operations and software installation.
Course Objectives: 1. Introduce students to the Android ecosystem, its architecture, and development tools. 2. Equip students with the skills to design and implement basic Android applications. 3. Teach students how to build interactive user interfaces with layouts, views, and navigation. 4. Introduce data storage concepts using SharedPreferences and SQLite. 5. Enable students to integrate learned concepts to develop a simple, functional Android application.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Install and configure the Android development environment.
CO 2: Design and develop a simple Android app with a user-friendly interface.
CO 3: Design and develop a simple Android app with a user-friendly interface.
CO 4: Create, test, and finalize a basic Android app that demonstrates the integration of learned concepts.

Contents

Sr. No	Title	Practical Hours
1	Install and configure Android Studio and Java JDK, Create the first project (Example: Expense Tracker App). Explore the Android Studio interface and understand the project structure.	02
2	Understand the basic structure of an Android app. Create and run a "Hello World" app on an emulator and physical device. Modify the app name and launcher icon for branding.	02
3	Introduction to XML layouts: LinearLayout and RelativeLayout. Design a simple app layout with TextView and EditText.	02
4	Add interactive elements: Add Buttons to the layout and handle button clicks using Java. (Example: Add input fields (EditText) for amount, category, and date. Arrange them using LinearLayout for a clean design.	02
5	Create a login form with TextView, EditText, and Button. Display input data using a Toast message. Implement Buttons and handle click events to navigate between screens. (Example: Create a second screen to display expense details.)	02
6	Display a Toast message when a button is clicked (e.g., "Expense Added Successfully"). Add validation to the expense form and show appropriate Toast messages like – “Please enter a valid amount.", "Category cannot be empty.", etc.	02

7	Learn how to create a splash screen that displays the app logo and name briefly before transitioning to the main activity.	02
8	Build a form and validate user inputs (Example: form for adding expenses, ensure amount is a number and category is not empty).	02
9	Learn about RecyclerView to display a list of items (Example: A static list of expenses in the app). Customize RecyclerView with images and text.	02
10	Customize RecyclerView with a layout for each data item (Example: amount, category, and date).	02
11	Introduction to SharedPreferences for storing user preferences. (Example: Save and retrieve the user's preferred expense currency like in USD, INR, etc.)	02
12	Add a Settings screen where users can update and save preferences.	02
13	Introduction to SQLite. Create a database and table for storing user's records. (Example: expense records.)	02
14	Implement functions to insert and retrieve data. Save user-entered data into the SQLite database. (Example: save expenses in the database)	02
15	Implement filtering options to view information by category or date. Create dropdown menus (Spinner) for selecting filters.	02

Final Project Assembly

1. Integrate all features:
2. Add expenses.
3. View and filter expenses.
4. Save data persistently using SQLite.
5. Display user preferences (currency).
6. Test the app on a real device and emulator.
7. Finalize the UI and fix bugs.

Text Books:

1. Android Programming for Beginners: Build in-depth, full-featured Android apps starting from zero programming experience, 3rd Edition, 2021, John Horton, Packt Publishing.
2. Android App Development for Dummies, 3ed, 2015, Michael Burton, Wiley

References Books:

1. Android Studio 4.1 Development Essentials - Kotlin Edition: Developing Android 11 Apps Using Android Studio 4.1, Kotlin and Android Jetpack, 1st Edition, 2021, Neil Smyth,PacktPublishing.
2. Android Programming: The Big Nerd Ranch Guide, 5th Edition, 2022, Bill Phillips, Chris Stewart, Kristin Marsicano, and Brian Gardner, Addison-Wesley Professional.

Online References:

1. NPTEL Course: Introduction to Modern Application Development, https://archive.nptel.ac.in/courses/106/106/106106156/
2. NPTEL Course: Mobile Computing, https://archive.nptel.ac.in/courses/106/106/106106147/

Semester: SIXTH

Syllabus
Semester VI

Course code:CAS41MM L304 **Course name:** Software Project & Agile Development **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:**L-2 P-1 **Evaluation scheme: 30**
Pre-requisites: Basic knowledge of software development processes and working experience within a team-based environment. Exposure to traditional project management methodologies
Course Objectives: 1. Deep understanding of Agile principles and methodologies 2. Working in team under Agile Frameworks and methodologies 3. Understand the role of Agile leadership and servant leadership 4. Guide learners in Agile Process and Frameworks with continuous improvements.
Course Outcomes: After completion of the course the student will be able to
CO 1: Students will start to manage software development effectively
CO 2: Create, Manage, and facilitate Agile Team to enhance collaboration
CO 3: Improve project delivery through continuous feedback and iterative development.
CO 4: Able to generate a demonstrative and adaptive environment that project needs in an organizational environments

Contents

Unit	Content	Teaching hours
1	**Introduction to Agile & Agile Project Management :**Agile principles & values, Traditional vs. Agile project management **,**Agile frameworks:, Agile Manifesto, Adopting Agile Mindset, Benefits of Agile, Agile Lifecycle **Agile Roles and Responsibilitie**:Agile /Scrum Roles, Agile responsibilities, Comparisons on Agile Roles	6
2	**Agile Frameworks & Methodologies:**Overview of Agile frameworks, Scrum framework: Roles, artifacts, ceremonies, Overview of Sprint, Kanban: Flow management, Kanban Board, Work In Process(WIP) limits, Lean principles & Agile scaling frameworks, Scaling Agile for Large Enterprise, Large Scale Scrum, Extreme Programming (XP) – Agile for Developers, Disciple Agile Delivery	8
3	**Agile Planning, Estimation & Execution:**AGILE EPICS, Features, User stories and Tasks, Framework under EPICS Backlog management & User Stories, Agile estimation: User Story Points, Planning, Sprint planning & estimation .**Agile Leadership, Collaboration & Stakeholder Management :**Handling Team Conflicts & Agile Coaching, Agile Roles: Product Owner, Scrum Master, Developers, Team Collaboration in Agile Leadership with Agile Roles: Product Owner, Scrum Master, Developers, Stakeholder Communication & Expectation Management	8

4	**Agile Metrics, Continuous Improvement & Certifications** Agile Metrics – Measuring Success in Agile Projects, Velocity, BurndownChart,Burnup Chart, Cycle Time,Lead Time, Escaped Defects,Cumulative Flow Diagram (CFD), Continuous Improvement – Retrospectives & Process Enhancement, Agile Retrospective - Importance , Types of Retrospective **Mad, Sad, Glad** – Understanding team emotions, **Start, Stop, Continue** – Actionable improvements, **Sailboat Retrospective** – Identifying risks and goals **Agile Certifications- Career aspects:** **Brief - Importance of Certificates career base.** - PMP, Certified Scrum Master (CSM),Professional Scrum, Master (PSM I, II, III),PMI Agile Certified Practitioner (PMI-ACP) Kanban Management Professional (KMP) , ICAgile Certified Professional (ICP)	8

Text Books: 1. Agile Software Development, Principles, Patterns, and Practices-Robert C. Martin Pearson Publications
2. Software Project Management- Hughes Bob- McGraw Hill Education Imprint
Reference Books: 1. The Art of Agile Development- James shore & Shane Warden -O'reilly Publications
2. Agile Project Management - J. Ross - BPB Publications
Online Resources: 1. NPTEL / SWAYAM lectures.
2. https://www.pmi.org/certifications/agile-certifications
3. https://www.atlassian.com/agile/kanban

Syllabus
Semester VI

Course code:CAS41MM P303 **Course name: Practical based on** Software Project & Agile Development **Course category:** Major Mandatory
Credits: 1 **Teaching scheme:** L-0 P-1 **Evaluation scheme:**
Pre-requisites: Basic knowledge of software development processes and working experience within a team-based environment. Exposure to traditional project management methodologies
Course Objectives: 1.Deep understanding of Agile principles and methodologies 2.Working in team under Agile Frameworks and methodologies 3.Understand the role of Agile leadership and servant leadership 4.Guide learners in Agile Process and Frameworks with continuous improvements.
Course Outcomes: After completion of the course the student will be able to
CO 1: Students will start to manage software development effectively
CO 2: Create, Manage, and facilitate Agile Team to enhance collaboration
CO 3: Improve project delivery through continuous feedback and iterative development.
CO 4: Confidently conduct stand-ups, sprint reviews, and retrospectives

Contents

Series	Observations & Analysis to done using Case studies (as per course coordinator)	Teaching hours
1	Project Charter Creation with Agile Mindset & Principles ● Define a project vision, objectives, stakeholders, scope, and constraints. ● Create a charter document outlining roles and responsibilities. ● Compare and Explore Agile values from the Agile Manifesto.	2
2	Work Breakdown Structure (WBS) Development ● Break the project into smaller, manageable tasks. ● Implement as per Agile works and how it improves software development. ● Use tools like MS Project, Trello, or Jira to visualize tasks.	2
3	Project Scheduling with Gantt Chart ● Develop a timeline using Gantt charts in MS Project or online tools. ● Identify task dependencies and milestones	2
4	Implement Agile Frameworks in Real-World Projects ● Hands-on application of Scrum, Kanban, and Lean principles use tool like JIRA ● Work with Sprint planning, user stories, and iterative development.	2
5	Use Agile Tools & Techniques in Live Projects ● Set up a Scrum board in Jira, Trello, or Azure DevOps. ● Manage and understand backlog, sprints, and task tracking in an Agile tool.	2

6	Develop Estimation & Planning Skills ● Use real techniques like Story Points, Planning Poker, and Velocity tracking. ● Perform backlog refinement and sprint forecasting.	2
7	Facilitate Agile Ceremonies ● Practice conducting Daily Stand-ups, Sprint Reviews, and Retrospectives. ● Learn techniques for improving team collaboration.	2
8	Manage Agile Teams & Stakeholders ● Lead cross-functional teams in an Agile environment. ● Handle stakeholders' expectations using Agile reporting tools.	2
9	Adapt Agile in Different Business Scenarios ● Work with Agile in startups, enterprises, and hybrid project management setups. ● Learn how to integrate Agile with DevOps.	2
10	Handle Agile Challenges & Continuous Improvement ● Learn to manage scope creep, risk, and impediments in Agile projects. ● Focus on continuous learning and delivery improvement.	2

Text Books:

1. Agile Software Development, Principles, Patterns, and Practices - Robert C. Martin - Pearson Publications
2. Software Project Management - Hughes, Bob - McGraw-Hill Education Imprint

Reference Books:

1. The Art of Agile Development - James Shore & Shane Warden - O'Reilly Publications
2. Agile Project Management - J. Ross - BPB Publications

Online Resources:

1. NPTEL / SWAYAM lectures.
2. https://www.pmi.org/certifications/agile-certifications
3. https://www.atlassian.com/agile/kanban

Syllabus
Semester VI

Course code: CAS41MML305 **Course name:** Advance Java **Course category:** Major Mandatory
Credits: 2 **Teaching scheme:** L-2 P-0 **Evaluation scheme:** CA – 30, ESE – 20
Pre-requisites: Proficiency in fundamental Java programming concepts, including object-oriented principles, core syntax, control structures, & exception handling.
Course Objectives: ● To understand and implement the concepts of interfaces and multiple inheritance in Java for designing modular and reusable applications. ● To explore and utilize Java packages for organizing and managing projects. ● To develop proficiency in multithreaded programming to handle concurrent processes and improve application performance. ● To design and deploy interactive applets using Java's applet programming capabilities.
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO 1: Demonstrate the ability to implement and integrate Java interfaces and multiple inheritance to develop modular programs.
CO 2: Effectively create and manage packages to structure and organize Java projects.
CO 3: Design and implement multithreaded programs to manage concurrent tasks and synchronization.
CO 4: Develop and deploy Java applets to create interactive web-based applications.

Contents

Unit	Content	Teaching hours
1	**INTERFACES: MULTIPLE INHERITANCE:** Introduction, defining interfaces, extending interfaces, implementing interfaces, accessing interface variables. **PACKAGES:** Putting Classes Together: Introduction, Java API Packages, Using System Packages	06
2	Naming Conventions, Creating Packages, Accessing a Package, Using a Package, Adding a Class to a Package, Hiding Classes, Static Import **MULTITHREADED PROGRAMMING:** Introduction, Creating threads, Extending the Thread Class, stopping and blocking a thread,Life Cycle of a thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing a runnable Interface.	08
3	**INTRODUCTION TO JAVA SERVLETS**: Introduction to Web Applications, Role of Servlets in Web Development, Difference Between Static and Dynamic Web Pages, Servlets vs. CGI (Common Gateway Interface), *Servlets Basics:* Understanding the	08

	Servlet API, Lifecycle of a Servlet (init(), service(), destroy()), Creating and Deploying a Simple Servlet, Configuring Servlets in web.xml	
4	*Handling Requests and Responses:* Understanding HttpServletRequest and HttpServletResponse, Handling GET and POST Requests, Reading Form Data from HTML, Sending Responses to the Client. *Servlet Communication and Session Management:* Request Dispatcher and Redirecting Requests, Cookies and Session Tracking, Managing User Sessions with HttpSession, URL Rewriting	08

Text Books:

1. Programming with Java – A Primer, 3rd Edition, E. Balagurusamy, Mc-Graw Hill
2. Java: The Complete Reference, 12th Edition, Herbert Schildt

Reference Books

1. Teach Yourself Java in 21 Days, 1st Edition, Laura Lemay, Charles L. Perkins, Sams.net Publishing
2. The Java Programming Language, 4th Edition, Ken Arnold, James Gosling, David Holmes, Addison Wesley Professional Publishers

Online Resources:

1. Programming In Java, By Prof. DebasisSamanta, IIT, kharagpurhttps://onlinecourses.nptel.ac.in/noc25_cs57/preview
2. Advanced Java Full Course, Youtube: https://www.youtube.com/watch?v=Ae-r8hsbPUo

Syllabus
Semester VI

Course code: CAS41M MP305 **Course name**: Practical based on Advanced JAVA **Course category**: Major Mandatory
Credits: 1 **Teaching scheme:**L-0 P-2 **Evaluation scheme:**CA-30 ESE-20
Pre-requisites: Basic understanding of Java programming concepts, including object-oriented principles, syntax, control structures, exception handling, file I/O, and basic multithreading and applet fundamentals.
Course Objectives: ● To provide hands-on experience in implementing Java interfaces to achieve multiple inheritance and modular programming. ● To enable students to create and use Java packages for effective organization and reuse of code. ● To develop practical skills in multithreaded programming to manage concurrent tasks and ensure synchronization in Java. ● To equip students with the ability to design and deploy interactive applets and integrate them with web pages.
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Demonstrate proficiency in defining, extending, and implementing interfaces for creating reusable and scalable applications**.**
CO 2: Apply concepts of package creation and usage to efficiently manage Java projects.
CO 3: Develop and execute multithreaded Java programs to handle concurrent processes with proper synchronization.
CO 4: Create interactive applets, understand their lifecycle, and integrate them into HTML web pages effectively.

Contents

Sr. No	Title	Practical Hours
1	Write a program to define and implement an interface, showcasing multiple inheritance by combining methods from different interfaces.	02
2	Extend an existing interface and demonstrate how methods from both the parent and child interfaces can be used in a single class.	02
3	Accessing interface variables: Define an interface Constants with a variable PI = 3.14. Create a class Circle that uses this constant to calculate the area of a circle.	02
4	Create and organize classes into packages, demonstrating how to access and utilize classes from different packages.	02
5	Implement a program to showcase the use of system packages (likejava.util or java.io) and custom packages in the same project.	02
6	Hiding classes in a package: Create a package library with two classes Book and Librarian. Restrict access to Librarian and demonstrate it by trying to access it from another class.	02

7	Write a program to create and manage multiple threads, ensuring proper synchronization where required.	02
8	Develop a program to explore thread lifecycle methods such as starting, stopping, and pausing threads, as well as handling thread priorities.	02
9	Create threads by extending the Thread class: Create a program that spawns two threads. The first thread prints numbers from 1 to 10, and the second thread prints numbers from 10 to 1.	02
10	Design a simple applet that demonstrates the applet lifecycle stages, including initialization, running, and termination.	02
11	Create an interactive applet that accepts user input and dynamically updates the applet's display based on user interaction	02

Text Books:

1.Programming with Java – A Primer, 3rd Edition, E. Balagurusamy, Mc-Graw Hill
2.Java: The Complete Reference, 12th Edition, Herbert Schildt

Reference Books

1.Teach Yourself Java in 21 Days, 1st Edition, Laura Lemay, Charles L. Perkins, Sams.net Publishing
2.The Java Programming Language, 4th Edition, Ken Arnold, James Gosling, David Holmes, Addison Wesley Professional Publishers

Online Resources:

1.Programming In Java, By Prof. DebasisSamanta, IIT, kharagpurhttps://onlinecourses.nptel.ac.in/noc25_cs57/preview
2.Advanced Java Full Course,
Youtube: https://www.youtube.com/watch?v=Ae-r8hsbPUo

Syllabus
Semester VI

Text Books:

1. William Stallings, "Crpyptography and Network security Principles and Practices", Pearson/PHI.
2. Wade Trappe, Lawrence C Washington, " Introduction to Cryptography with coding theory", Pearson.

Reference Books:

1.Behrouz A Forouzan, DebdeepMukhopadhyay, "Cryptography And Network Security", McGraw Hill Education, 3rd Edition..

2.Kaufman, C., Perlman, R., &Speciner, M. (2006). Network security: Private communication in apublic world (2nd ed.). Prentice-Hall/Pearson.

Online Resources:

1.https://onlinecourses.nptel.ac.in/noc25_ee54/previewhttps://onlinecourses.nptel.ac.in/noc22_cs90/preview

Course code: CAS41MML306 **Course name:** Network Security **Course category:** Major Mandatory

Credits:2 **Teaching scheme**: L-2 **Evaluation scheme:** CA–30, ESE–20

Pre-requisites: An understanding of the basics of computer networking and programming.

Course Objectives: To understand the concepts of information security in a local area network and the Internet

Course Outcomes: After completion of the course the student will be able to demonstrate:

CO 1: To learn the vulnerabilities of computer networks to attacks by adversaries and hackers

CO 2: To evaluate the methods and techniques to defend against these attacks and to minimize their damage

CO 3: To identify and analyze security problems in computer systems and networks.

CO 4: To develop security mechanisms to protect computer systems and networks.

Contents

Unit	Content	Teaching hours
1	**INTRODUCTION TO SECURITY & SYMMETRIC ENCRYPTION- Security Concepts:** Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security**. Conventional Encryption:** Conventional encryption model -	06

	classical encryption techniques -substitution ciphers and transposition ciphers – cryptanalysis – steganography - stream and blockciphers - Modern Block Ciphers	
2	**PUBLIC KEY CRYPTOGRAPHY AND AUTHENTICATION REQUIREMENTS:**-Principles of public key crypto systems - RSA algorithm - security of RSA - key management – Diffle-Hellman key exchange algorithm **Authentication requirements** - authentication functions - message authentication code - hash functions - birthday attacks – security of hash functions and MACS.	08
3	**INTEGRITY CHECKS AND AUTHENTICATION ALGORITHMS**: MD5 message digest algorithm - Secure hash algorithm (SHA) Digital Signatures: Digital Signatures - authentication protocols - digital signature standards (DSS) - proof of digital signature algorithm - Authentication Applications: Kerberos and X.509 - directory authentication service - electronic mail security-pretty good ,privacy (PGP) - S/MIME.	08
4	**IP SECURITY AND KEY MANAGEMENT:IP Security:** Architecture - Authentication header - Encapsulating security payloads - combining security associations - key management. **WEB AND SYSTEM SECURITY: Web Security:** Secure socket layer and transport layer security - secure electronic transaction (SET) - **System Security**: Intruders - Viruses and related threads - firewall design principles – trusted systems.	08

Syllabus
Semester VI

Course code: CAS41MEL303	**Course name:** AWS DevOps	**Course category:** Major Elective
Credits: 03	Teaching scheme: L-3 P-0	Evaluation scheme: CA–30, ESE–20
Pre-requisites: : Basic knowledge of Cloud computing, Linux operating system.		
Course Objectives: To learn and develop Concept of DevOps		
Course Outcomes: After completion of the course the student will be able to demonstrate:		
CO 1: Understanding the basics of cloud computing.		
CO 2: To know the Cloud Computing Web Services and its Platforms.		
CO 3: To deploy the Cloud computing services.		
CO 4: Basics of AWS, Docker and Kubernetes.		

Contents

Unit	Content	Teaching hours
1.	**Introduction to DevOps:** Basic Concept of DevOps, Benefits of DevOps in modern software development, DevOps lifecycle: CI/CD, Monitoring, and Automation, Role of AWS in DevOps. **Introduction to AWS:** Overview of AWS services, Setting up an AWS account, Introduction to AWS Management Console, Key services for DevOps, EC2, S3, IAM, Lambda, RDS, and Cloud Formation.	06
2.	**Version Control with Git and GitHub:**Git Basics: Repositories, branching, merging, and commits, Using GitHub for collaboration, Setting up and managing code repositories. **AWS Elastic Compute Cloud (EC2):** Setting up EC2 instances, Configuring SSH access and security groups, Managing servers with EC2	08
3.	**Infrastructure as Code (IaC):** Introduction to AWS CloudFormation, Writing CloudFormation templates, Deploying and managing infrastructure using IaC, Introduction to Terraform with AWS. **Continuous Integration and Continuous Delivery (CI/CD): O**verview of CI/CD pipelines, Introduction to AWS CodePipeline, Integrating AWS CodeCommit, CodeBuild, and CodeDeploy, Automating deployments to EC2 and Lambda.	08

4.	**Containerization with Docker:** Introduction to Docker and containerization, Building and managing Docker containers, Dockerizing applications for AWS. **Container Orchestration with Kubernetes and AWS EKS:** Overview of Kubernetes, Setting up Kubernetes clusters using Amazon EKS, Deploying and managing containerized applications in Kubernetes.	08

Text Books:

1. A Complete Guide to DevOps with AWS By Osama Mustafa, Apress Berkeley, CA, 1st Edition.
2. Engineering DEVOPS By Marc Hornbeek.
3. DevOps by Sanjeev Sharma and Bernie Coyne, AWS Well-Architected, 2nd IBM Limited Edition.

Reference Books:

1. Effective DevOpsBy Jennifer Davis and Ryn Daniels O'Reilly.

Online Resources:

1. NPTEL/SWAYAM courses

Syllabus
Semester VI

Course code: CAS41M EP303 **Course name:** Practical Based on AWS DevOps **Course category:** Major Elective
Credits: 1 **Teaching scheme**: L-0 P-2 **Evaluation scheme:** CA–30**, ESE–**20
Pre-requisites: Basic knowledge of Cloud computing, Linux operating system.
Course Objectives: : To learn and develop Concept of DevOps
Course Outcomes: At the end of the course, the students will be able to –
CO 1: Understanding the basics of cloud computing.
CO 2: To know the Cloud Computing Web Services and its Platforms.
CO 3: Basics of AWS, Docker and Kubernetes.

Contents

Sr. No	Title	Practical Hours
1	Create and configure an AWS account.	02
2	Set up billing alarms to monitor costs.	02
3	Launch and connect to an EC2 instance via SSH.	02
4	Configure security groups for access & Set up a web server (e.g., Apache or Nginx) on EC2.	02
5	Create and configure a Virtual Private Cloud (VPC).	02
6	Create an S3 bucket and upload/download files.	02
7	Configure S3 bucket policies for public/private access & Enable versioning and lifecycle rules in S3.	02
8	Write a CloudFormation template to deploy an EC2 instance.	02
9	Create a repository in AWS CodeCommit.	02
10	Set up a pipeline using AWS CodePipeline to deploy a sample application.	02
11	Automate builds using AWS CodeBuild& Deploy applications with AWS CodeDeploy.	02

12	Install Docker and create a Dockerfile for a sample application.	02
13	Build and run a Docker container locally & Push the Docker image to Amazon Elastic Container Registry (ECR).	02
14	Create an EKS cluster.	02
15	Create and manage IAM users, groups, and roles.	02

Text Books:

1. A Complete Guide to DevOps with AWS By Osama Mustafa, Apress Berkeley, CA, 1st Edition.
2. Engineering DEVOPS By Marc Hornbeek.
3. DevOps by Sanjeev Sharma and Bernie Coyne, AWS Well-Architected, 2nd IBM Limited Edition.

References Books:

Effective DevOps By Jennifer Davis and Ryn Daniels O'Reilly.

Online References:

NPTEL/SWAYAM course

Syllabus
Semester VI

Course code: CAS41MEL304 **Course name:** Data Mining and Visualization **Course category:** Major Elective
Credits: 3 **Teaching scheme:** L-0 P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Students should know the basic concepts of data Structure and analysis
Course Objectives: Students will be able to actively manage and participate in data mining projects. To develop research interest towards advances in data mining. Students will be able to understand the visualization techniques
Course Outcomes: After completion of the course the student will be able to demonstrate:
CO1: Identify appropriate data mining algorithms to solve real world problems.
CO2: Compare and evaluate different data mining techniques like classification, prediction, clustering and association rule mining.
CO3: Describe complex data types with respect to spatial and Data Visualization.
CO4: Benefit the user experiences towards research and innovation. Integration in the Data Mining area.

Contents

Unit	Content	Teaching hours
1	**Introduction to Data Mining:** Why Mine Data, Commercial Viewpoint, Scientific Viewpoint Motivation, Definitions, Origins of Data Mining, Data Mining Tasks, Classification, Clustering, Association Rule Discovery, Sequential Pattern Discovery, Regression, Challenges of Data Mining, Data Mining Data: What is Data? Attribute Values, Measurement of Length, Types and Properties of Attributes, Discrete and Continuous Attributes, Types of data sets, Data Quality, Data Preprocessing, Aggregation, Sampling, Dimensionality Reduction, Feature subset selection, Feature creation, Discretization and Binarization, Attribute Transformation, Density.	10
2	**Data Mining:** Exploring Data, Data Exploration Techniques, Summary Statistics, Frequency and Mode, Percentiles, Measures of Location: Mean and Median, Measures of Spread: Range and Variance, Visualization, Representation, Arrangement, Selection, Visualization Techniques: Histograms, Box Plots, Scatter Plots, Contour Plots, Matrix Plots, Parallel Coordinates, Other Visualization Techniques, OLAP : OLAP Operations	10
3	**Data Mining Classification:**	10

	Basic Concepts, Decision Trees, and Model Evaluation: Classification: Definition, Classification Techniques, Tree Induction, Measures of Node Impurity, Practical Issues of Classification, ROC curve, Confidence Interval for Accuracy, Comparing Performance of Two Models, Comparing Performance of Two Algorithms	
4	**Data Mining Classification:** Alternative Techniques: Rule-Based Classifier, Rule Ordering Schemes, Building Classification Rules, Instance-Based Classifiers, Nearest Neighbor Classifiers, Bayes Classifier, Naive Bayes Classifier, Artificial Neural Networks (ANN), Support Vector Machines.	10
5	**Introduction to Data Visualization:** Classification of Visualization techniques – Structure and representation, Selection of a Visualization, Visualizations for high dimensional data, Graphics sand computing, Principles of Data Visualization, Multivariate data, Linked data, Visualizing trees and forests, Large Datasets – Plots and their variates, Visualizing cluster analysis ,contingency tables – finite mixture models, Methodologies: Visualization in Bayesian data analysis, Matrix visualization, Data visualization by kernel machines .Applications Visualization for genetic network reconstruction, medical images, financial dataset and Insurance risk processes.	05

Text Books: Tan, Steinbach, Kumar. Introduction to Data Mining
Reference Books: Jiawei Han, Micheline Kamber Data Mining: Concepts and Techniques Morgan Kaufmann Publishers
Online Resources: NPTEL / SWAYAM lectures

Syllabus
Semester VI

Course code: CAS41M EP304 **Course name**: Practical Based on Data Mining and Visualization **Course category**: Major Elective
Credits:1 **Teaching scheme:** L-0 P-0 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basics concepts of Data Structure
Course Objectives: Students will be able to actively manage and participate in data mining projects. To develop research interest towards advances in data mining. Students will be able to understand the visualization techniques
Course Outcomes: At the end of the course, the students will be able to –
CO1: Identify appropriate data mining algorithms to solve real world problems.
CO2: Compare and evaluate different data mining techniques like classification, prediction, clustering and association rule mining.
CO3: Describe complex data types with respect to spatial and Data Visualization.
CO4: Benefit the user experiences towards research and innovation. Integration in the Data Mining area.

Contents

Sr. No	Title	Practical Hours
1	Demonstration of preprocessing on dataset student.arff	01
2	Demonstration of preprocessing on dataset labor.arff	01
3	Demonstration of Association rule process on dataset contactlenses.arff using apriori algorithm	01
4	Demonstration of Association rule process on dataset test.arff using apriori algorithm	01
5	Demonstration of classification rule process on dataset using Nearest neighbor algorithm	01
6	Demonstration of classification rule process on dataset using K-NN algorithm	01
7	Demonstration of classification rule process on dataset using Decision tree algorithm	01
8	Demonstration of classification rule process on dataset using Regression algorithm	01
9	Apply Visualization techniques for Various Dataset	01
10	Apply Visualization techniques for Various Dataset	01

Text Books:
1.Tan, Steinbach, Kumar.Introduction to Data Mining
References Books:
1.Jiawei Han, MichelineKamberData Mining: Concepts and TechniquesMorgan Kaufmann Publishers
Online References: NPTEL/ SWAYAM online courses

Semester: SEVEN

Syllabus
Semester-VII

Course Code: CAS41MML401 **.**	**Course Name:** Software Testing and Quality Assurance **Course Category:** Major Mandatory
Credits: 3	**Teaching Scheme:** L-3 P-0 **Evaluation Scheme:** CA-60, ESE-40
Pre-requisites: Basic programming and understanding of SDLC	
Course Objectives: Learn software testing fundamentals, design and execute test cases, perform various testing types, and apply SQA practices using tools.	
Course Outcomes: At the end of the course, the students will be able to -	
CO1: Understand the fundamentals of software testing, STLC, testing objectives, and test case design.	
CO2: Apply black box and white box testing techniques, coverage methods, and object-oriented testing concepts.	
CO3: Identify testing levels, types, and approaches, and differentiate between verification & validation and manual & automated testing.	
CO4: Understand software quality assurance concepts, quality models, metrics, standards, and reliability measures.	
CO5: Gain knowledge of computer-aided testing tools and review techniques used in software testing and quality assurance.	

Course Contents –

Unit	Content	Teaching hours
1	**Unit – I: Introduction to Software Testing and Test Design** Introduction to software testing and need of software testing. Error, fault and failure. Testing fundamentals and testing objectives. Test information flows. Software testing life cycle (STLC). Validation testing activities. Low-level testing and high-level testing. Test cases and test case designing techniques.	5
2	**Unit – II: Testing Techniques and Coverage** Black box testing and white box testing. Functional testing. Equivalence partitioning. Boundary Value Analysis (BVA). Decision table–based testing. Cause–effect graphing. Syntax testing. Structural testing and coverage testing. Statement coverage. Branch and decision coverage. Path coverage. Cyclomatic complexity. Black box versus white box testing.Object-oriented testing: issues in object-oriented testing, class testing and GUI testing.	10
3	**Unit – III: Levels, Types and Approaches of Testing** Verification and validation. Levels of testing: unit testing, integration testing, system testing and acceptance testing. Alpha testing and beta testing. Static testing versus dynamic testing. Manual testing versus automatic testing. Tester's workbench and eleven (11) steps of testing process. Types of testing: installation testing, usability testing, regression testing, performance testing, load testing, stress testing and security testing. Static and dynamic testing techniques. Static testing techniques.	10

4	**Software Quality Assurance**: Software Quality: definition of quality and quality factors, Quality Assurance (QA) and Software Quality Assurance (SQA), Need for SQA, Goals of SQA, Elements and building blocks of SQA, SQA tasks and activities, Process quality and product quality, Attributes of software quality and software quality metrics, Formal approaches to SQA, Statistical Software Quality Assurance, Software reliability, Measures of reliability and availability, Software safety, SQA planning and standards, SQA plan, ISO 9000 quality standards, SEI Process Capability Maturity Model (CMM), Six Sigma for software engineering.	10
5	Computer-Aided Software Testing Tools (CAST): Static Testing Tools, Dynamic Testing Tools, and Characteristics of Modern Tools. e.g. Win Runner, Load Runner, Rational ROBO. **Review types:** Informal Review, Technical or peer review, Walkthrough, Inspection, static analysis, Review meeting and reporting, Review guidelines, Review checklist.	10

Text Books: 1.Software Engineering – A Practitioners Approach Roger S. Pressman, 4th /7th Edition,Mcgraw Hill, International Education. 2. An Integrated Approach To S/w Engineering, Pankaj Jolote, 1st / 2nd Edition, Narosa
Reference Books: 1. Software Engineering – A Programming Approach, D. Belie I. Moray, J. Rough, PHI. 2. Software Testing Techniques, Barrios Bier, 2nd Edition, Van N Ostrand Reinhold. 3. Software Engineering Concepts-Richard Fairley, CDAC. Tata McGraw-Hill Series**.**
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus
Semester-VII

Course code: CAS41MML402		**Course Name**: Blockchain Technology **Course Category**: Major Mandatory
Credits:3	**Teaching scheme:** L-3 P-0	**Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic knowledge of computers, programming concepts, and computer networks.		
Course Objective: The objective of this course is to introduce students to the fundamentals of blockchain technology, cryptography, consensus mechanisms, smart contracts, and real-world applications of decentralized systems.		
Course Outcome: After completion of the course the student will be able:		
CO1. Understand the basic concepts, architecture, and components of blockchain technology.		
CO2. Explain cryptographic techniques and consensus mechanisms used in blockchain systems.		
CO3. Analyze Bitcoin and Ethereum blockchain platforms.		
CO4. Develop and deploy basic smart contracts using Solidity.		
CO5. Identify enterprise blockchain solutions and real-world use cases.		

Contents:

Unit	Contents	Teaching Hours.
1	**Introduction to Blockchain:** Basics of blockchain and distributed ledger technology, centralized vs decentralized vs distributed systems, history and evolution of blockchain, features and characteristics of blockchain, applications and advantages of blockchain technology.	**09**
2	**Blockchain Architecture and Cryptography:** Blockchain structure, blocks and block headers, genesis block, hashing and chaining mechanism, types of blockchains, basics of cryptography, hash functions (SHA-256), public and private key cryptography, digital signatures, Merkle trees and data integrity.	**09**
3	**Consensus Mechanisms and Bitcoin:** Need for consensus, Byzantine Generals Problem, Proof of Work, Proof of Stake, Delegated Proof of Stake, Proof of Authority, introduction to Bitcoin, Bitcoin architecture, wallets, UTXO model, Bitcoin transactions, mining process, mining pools and incentives.	**09**

4	**Ethereum and Smart Contracts:** Introduction to Ethereum, Ethereum ecosystem, Ether and gas, Ethereum Virtual Machine, smart contracts, Solidity basics including data types, variables, functions, mappings and modifiers, smart contract lifecycle, deployment using Remix and MetaMask, ERC-20 tokens and ERC-721 NFTs.	**09**
5	**Enterprise Blockchain and Applications:** Permissioned and permissionless blockchains, introduction to Hyperledger Fabric, peers, orderers and channels, decentralized applications and DApp architecture, Web3 overview, Web3.js and Ethers.js, blockchain use cases in supply chain, healthcare, digital identity, banking and DeFi, challenges and future scope of blockchain technology.	**09**

Books and References:

Text Books **1.** Imran Bashir, *Mastering Blockchain*, Packt Publishing. **2.** Chandramouli Subramanian et al., *Blockchain Technology*, Universities Press.
Reference Books: **1.** Andreas M. Antonopoulos, *Mastering Bitcoin* and *Mastering Ethereum*, O'Reilly. **2.** Daniel Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*. **3.** Ritesh Modi, *Solidity Programming Essentials*.
Online Resources: **1.** Ethereum Documentation – https://ethereum.org/developers/ **2.** Solidity Documentation – https://docs.soliditylang.org/ **3.** Hyperledger Fabric Documentation – https://hyperledger-fabric.readthedocs.io/ **4.** NPTEL – Blockchain Architecture Design and Use Cases (IITs)

Syllabus
Semester-VII

Course code: CAS41MML403		**Course Name**: Python Programming **Course Category**: Major Mandatory
Credits:3	**Teaching scheme:** L-3 P-0	**Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic understanding of computer fundamentals and problem-solving concepts.		
Course Objective:		
To equip students with foundational knowledge and practical skills in Python programming, including control flow, data structures, functions, modules, and object-oriented programming concepts.		
Course Outcome: After completion of the course the student will be able:		
CO1. Develop python programs using control flow statements.		
CO2. Perform operations on various data structures in Python.		
CO2. Develop packages to solve given problem using python.		
CO3. Apply object-oriented approach to solve given problem using python.		

Contents:

Unit	Contents	Teaching Hours.
1	• **Introduction to Python and Control flow statements** Introduction: Features, History and Applications of Python, Python IDEs. Python building blocks: Indentation, Identifiers, Variable, Comments, Keywords. Basic input output operations: input (), print() Operators: Arithmetic, Relational, Assignment, Logical, Bitwise, Membership and Identity operator. Control flow statements: Conditional statements (if, if-else, if-elif-else, nested if), Loops in python (while, for, nested loops), Loop manipulation statements (continue, pass, break, else).	**10**
2	• **Data Structures in Python** List: a) Defining lists, accessing values from list, deleting list values, updating lists b) Basic list operations c) Built-in list functions/methods. Tuple: a) Defining Tuple, accessing values from Tuple b) Basic Tuple operations c) Built in Tuple functions/methods Set: a) Defining Sets, accessing values from set, deleting set values b) Basic set operations c) Built in set functions/methods. Dictionary: a) Defining Dictionary, accessing values from Dictionary, deleting Dictionary values, updating Dictionary b) Basic Dictionary operations c) Built in Dictionary functions/methods.	**10**

3	• **Functions, Modules and Packages in Python** Functions: Defining function, calling function, Function arguments, Return statement, Scope of Variable, Lambda functions. Modules: Create user defined Module, importing a module, Using python built-in modules, Namespace and scoping Python Packages: Create user defined Package, importing a Package, Using python built-in Packages.	**10**
4	• **Object Oriented Programming in Python** Object oriented Concepts: Creating class, Creating object Constructors in python (Parameterized & Nonparameterized), the self-parameter . Polymorphism: Method Overloading and Overriding. Data Hiding / Abstraction . Inheritance: Single Inheritance, Multiple Inheritance, Multilevel Inheritance.	**10**
5	• **Advanced Python Programming and Modules** Python Modules and Packages: Built-in modules, user-defined modules, importing modules, creating and using packages. Exception Handling in Python: Types of errors, try–except–else–finally blocks, raising exceptions, custom exceptions. File Handling in Python: File modes, reading and writing files, working with text and binary files, context manager (with statement). Working with Standard Libraries: File modes, reading and writing files, working with text and binary files, context manager (with statement).	**10**

Books and References:

Text Books 1.R. Nageswara Rao "Core Python Programming" Dreamtech Press, ISBN-13:9789390457151 2. David Amos, Dan Bader, Joanna Jablonski, Fletcher Heisler "Python Basics" Real Python, ISBN-13: 9781775093329.
Reference Books: 1.Rupesh Nasre, "Python Programming", AICTE, ISBN 9788195986354 [Online available on AICTE e-Kumbh] 2.Mark Lutz, "Learning Python", O'Reilly Media, Inc, ISBN: 9781449355739
Online Resources: 1.**Python for Beginners** - https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013094439793500160259 2_shared/overview 2. **Python Programming Lab**- https://spoken-tutorial.org/watch/Python+3.4.3/Input-output/ English/ 3.**OOPs using Python** - https://www.programiz.com/python-programming/objectoriented -programming

Syllabus
Semester-VII

Course Code: CAS41MMP401 **Course Name:** Practical Based on Software Testing and Quality Assurance . **Course Category:** Major Mandatory
Credits: 1 **Teaching Scheme:**L-0,P-2 **Evaluation Scheme**: CA-30, ESE-20
Pre-requisites: Basic programming, understanding of SDLC, and computer fundamentals.
Course Objectives: Learn software testing concepts, design and execute test cases, perform different testing types, and apply SQA practices using tools
Course Outcomes: At the end of the course, the students will be able to -
CO1: Understand basic software testing concepts and STLC.
CO2: Design test cases using black box testing techniques.
CO3: Apply white box testing and coverage analysis methods.
CO4: Perform different levels and types of software testing
CO5: Analyze software quality using SQA practices and testing tools

Note: Tools Used: LibreOffice Calc, Selenium, and pytest (Python) / JUnit (Java).

List of Practical's:

Sr.No.	Title of the Experiment	Practical Hours
1	Study of software testing concepts and Software Testing Life Cycle (STLC) and preparation of test case template.	2
2	Implement test cases using **Equivalence Partitioning** and **Boundary Value Analysis (BVA).**	2
3	Implement test cases using **Decision Table** technique.	2
4	Implement test cases using **Cause–Effect Graphing** technique.	2
5	Perform **Functional (Black Box) Testing** on a given software application.	2
6	Perform structural **(White Box) Testing** including statement and branch coverage.	2
7	Calculate **Cyclomatic Complexity** and identify independent paths for a given program.	2
8	Perform **Unit Testing and Integration Testing** of a software module.	2
9	Perform **Regression Testing and Usability Testing** on a software application.	2
10	Perform **Performance and Load Testing** of an application and Study of SQA activities, quality metrics, ISO 9000, CMM, and Six Sigma.	2
11	Project	10

Text Bosok:1.Srinivasan Desikan & Gopalaswamy Ramesh – *Software Testing: Principles and Practices*, Pearson
Reference Books: 1. Naresh Chauhan – *Software Testing*, Oxford University Press 2. Roger S. Pressman – *Software Engineering: A Practitioner's Approach*, McGraw-Hill
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus
Semester-VII

Course code: CAS41MMP402 **Course Name**: Practical based on Blockchain Technology **Course Category**: Major Mandatory
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of computers, programming fundamentals, and internet usage.
Course Objective: The objective of this practical course is to provide hands-on exposure to blockchain concepts, cryptographic techniques, smart contracts, and basic decentralized application development.
Course Outcome: After completion of the course the student will be able:
CO1. Perform basic cryptographic operations used in blockchain systems..
CO2. Implement a simple blockchain structure and validate blocks.
CO3. Use Ethereum tools such as MetaMask and Remix for smart contract deployment.
CO4. Develop and test basic smart contracts and tokens.
CO5. Understand decentralized applications and enterprise blockchain platforms.

Contents:

Sr. no.	Contents	Practical Hours.
1	Generate SHA-256 hash for different text inputs and implement a simple Merkle Tree using Java or Python.	**02**
2	Create a basic Blockchain class supporting block creation, proof-of-work, and chain validation.	**02**
3	Setup a MetaMask wallet, connect to an Ethereum test network, and perform a test Ether transaction.	**02**
4	Write and compile a simple “Hello World” smart contract using Solidity in Remix IDE.	**02**
5	Create a Bank Smart Contract allowing deposit, withdrawal, and balance checking.	**02**
6	Launch a custom ERC-20 token on a test network and transfer tokens between two accounts.	**02**
7	Develop a Voting DApp frontend using HTML and JavaScript interacting with a smart contract via Web3.js.	**02**
8	Set up a local private blockchain using Ganache and deploy a smart contract in a local environment.	**02**
9	Implement and analyze Ethereum token standards such as ERC-20 and ERC-721 with real-world examples.	**02**
10	Design and analyze decentralized applications architecture and enterprise blockchain platforms with use cases.	**02**

Books and References:

Text Books **1.** Imran Bashir, *Mastering Blockchain*, Packt Publishing. **2.** Chandramouli Subramanian et al., *Blockchain Technology*, Universities Press.
Reference Books: **1.** Andreas M. Antonopoulos, *Mastering Bitcoin* and *Mastering Ethereum*, O'Reilly. **2.** Daniel Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*. **3.** Ritesh Modi, *Solidity Programming Essentials*.
Online Resources: **1.** Ethereum Documentation – https://ethereum.org/developers/ **2.** Solidity Documentation – https://docs.soliditylang.org/ **3.** Hyperledger Fabric Documentation – https://hyperledger-fabric.readthedocs.io/ **4.** NPTEL – Blockchain Architecture Design and Use Cases (IITs)

Syllabus
Semester-VII

Course code: CAS41MML403	**Course Name**: Practical Based on Python Programming **Course Category**: Major Mandatory	
Credits:1	**Teaching scheme:** L-0 P-2	**Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic understanding of computer fundamentals and problem-solving concepts.		
Course Objective:		
To equip students with foundational knowledge and practical skills in Python programming, including control flow, data structures, functions, modules, and object-oriented programming concepts.		
Course Outcome: After completion of the course the student will be able:		
CO1. Develop python programs using control flow statements.		
CO2. Perform operations on various data structures in Python.		
CO2. Develop packages to solve given problem using python.		
CO3. Apply object-oriented approach to solve given problem using python.		

List of Practical

Practical No.	Contents	Practical Hours
1	1. Install the given Python IDE. 2. Write python program to display welcome message on screen. 3. Implement the python program to read data from user and display data on screen	**2**
2	4. Implement a python program using following operators: 1. Arithmetic 2. Relational & logical 3. Assignment 5. Bitwise 6. Membership 7. Identity 4. Implement a python program to demonstrate the use of following conditional statements: 1. if statement 2. if..else statement 3. if..elif..else statement 4. nested if statement	**2**

3	5. Implement a python program to demonstrate the use of following looping statements: 1. while loop 2. for loop 3. nested loop 6. Implement python program to demonstrate the use of loop control statements. [continue, pass, break, else]	**2**
4	7. Implement a python program to perform following operations on the List: 1. Create a List 2. Access List 3. Update List **4.** Delete List 8. Implement Python program to demonstrate the use of built-in functions/methods on List (Any five Functions/methods)	**2**
5	9. Implement python program to perform following operations on the Tuple: 1. Create a Tuple 2. Access Tuple 3. Print Tuple 4. Delete Tuple 10. Implement a python program to perform following operations on the Dictionary: 1. Create a Dictionary 2. Access Dictionary 3. Update Dictionary 4. Delete Dictionary 5. Looping through Dictionary	**2**
6	11. Write a user define function to implement following features: 1. Function without argument 2. Function with argument 3. Function returning value 12. Write Python program to demonstrate use of following advanced functions: 1. lambda 2. map 3. reduce	**2**

7	13. Develop a python program to perform following operations: 1. Creating a Class with method 2. Creating Objects of class 3. Accessing method using object 14. Write a python program to demonstrate the use of constructors: 1. Default 2. Parameterized 3. Constructor Overloading	**2**
8	15. Implement a python program to demonstrate 1. Method Overloading 2. Method Overriding 16. Write a python program to implement 1. Single inheritance 2. Multiple Inheritance 3. Multilevel inheritance	**2**
9	17. Write a program to implement exceptional handling using python. 18. Write a program in python to implement the file handling operations in python.	**2**

Syllabus
Semester-VII

Course code: CAS41MEL401	**Course name:** React Java Script **Course category:** Major Elective
Credits: 3 Teaching scheme: L-3 P-0 **Evaluation scheme:** CA–60, ESE–40	
Pre-requisites: Knowledge of Object-Oriented Concepts, Database, HTML, CSS, JavaScript.	
Course Objectives: Building a strong foundation in modern JavaScript, core React concepts, state management, routing, and basic API integration. The curriculum is designed to enable students to build functional front-end applications.	
Course Outcomes: After completion of the course the student will be able to:	
CO1: Understand the fundamentals of React java Script.	
CO2: Implementation of web application using Node JS and Express JS.	
CO3: Understand Node.js architecture and core concepts.	

Contents:

Unit	Content	Teaching hours
1	**Introduction to React & Environment Setup**: Overview of React, its benefits (e.g., Virtual DOM), and setting up the development environment using tools like Node.js and create-react-app or Vite.	05
2	**Modern JavaScript (ES6+)**: Review of essential JavaScript features necessary for React, including arrow functions, let and const, destructuring, spread/rest operators, classes, promises, and modules. **Components and JSX**: Understanding the building blocks of React applications (functional vs. class components), the use of JSX syntax for templating, and component composition.	10

3	**State and Props**: Managing data within components using state and passing data between components using props, including validation with prop Types and default props. **Event Handling and Forms**: Handling user interactions with Reacts event system and managing form inputs through controlled components. **React Hooks**: In-depth coverage of fundamental hooks such as uses State, use Effect, use Context, use Ref, use Reducer, and creating custom hooks.	10
4	**Forms & User Input:** Handling form submission and user input using controlled components. **React Router:** Implementing client-side routing for single-page applications, including nested routes and dynamic navigation using components like Browser Router, Routes, Route, Link, and NavLink. **API Integration & Data Fetching:** Making HTTP requests to fetch or send data from external APIs using fetch or axios, and handling loading/error states.	10
5	**Styling:** Various approaches to styling React applications, including plain CSS, inline styles, CSS Modules, and popular frameworks like Bootstrap or Tailwind CSS. **Project and Deployment:** Building a simple CRUD application-Integrating frontend (React) with backend (Node.js and Express)-Deployment Strategies- Deployment options (e.g., Heroku, Netlify) - Deploying Node.js and React applications.	10

Textbooks:
1) Node.js Design Patterns: Design and implement production-grade Node.js applications using proven patterns and techniques, 3rd Edition.
2) The Road to React: The React.js with Hooks in JavaScript Book (2023 Edition)
Reference Books:
1) Beginning Node.js, Express & MongoDB Development.
2) Full stack React: The Complete Book on ReactJS - Full Package

Syllabus
Semester-VII

Course code CAS41M EL402 **Course Name**: Frontend Development (ASP.Net,Angular) **Course Category**: Major Elective
Credits: 3 **Teaching scheme:** L- 3 P - 0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: 1. Uunderstanding of **HTML, CSS, and JavaScript** **2.** Basic knowledge of **Web Development fundamentals**
Course Objective:
To equip students with the skills and knowledge required to design and develop dynamic, responsive, and interactive web applications using modern frontend technologies.
Course Outcome: After completion of the course the student will be able:
CO1. Design and implement responsive and user-friendly interfaces using HTML5, CSS3, and modern UI frameworks.
CO2. Develop robust single-page applications using the Angular framework, including components, services, routing, and state management.
CO3. Integrate Angular-based frontend applications with backend APIs developed using ASP.NET Core (from a frontend perspective)..
CO4: Apply best practices in frontend development, including performance optimization, security considerations, testing, and deployment.
CO5: Debug, troubleshoot, and maintain frontend and integrated full-stack web applications effectively.

Contents:

Unit	Contents	Teaching Hours.
I	**Introduction to Modern Frontend Development & Web Essentials**: Review of HTML5, CSS3, and JavaScript, Responsive web design principles; Introduction to CSS frameworks (e.g., Bootstrap, Material Design); Introduction to TypeScript—data types, interfaces, classes, modules; Overview of frontend frameworks and Single-Page Application concepts.	7
II	**Fundamentals of Angular** : Angular architecture; Data binding; Directives and pipes; Services and dependency injection; Routing and navigation; Template-driven and reactive forms; Introduction to RxJS and Observables for asynchronous operations.	10
III	**Advanced Angular Concepts & API Interaction** : State management techniques, Interceptors and route guards; RESTful API consumption using HttpClient module; CRUD operations; Authentication and authorization in SPAs; Performance optimization techniques for Angular applications.	9

IV	**ASP.NET Core for Frontend Integration** : Overview of .NET Core and ASP.NET Core MVC (focus on its role as a backend for frontend); Setting up an ASP.NET Core project to serve Angular applications; API development—Controllers, Actions, Routing, Model Binding; Data Transfer Objects; Introduction to Entity Framework Core (briefly); API security fundamentals.	9
V	**Deployment, Testing, and Best Practices** : Angular build and deployment process; Integrating Angular with ASP.NET Core for unified deployment; Unit testing in Angular; End-to-end testing concepts ; Version control and collaborative workflows; Introduction to CI/CD pipelines for web applications; Modern web security best practices for frontend and APIs.	10

Books and References:

Text Books

1. Freeman, Adam. *Essential Angular for ASP.NET Core MVC*. Apress, 2017.

2. Freeman, Adam. *Pro ASP.NET Core MVC*. Apress, latest edition.

Reference Books:

1. Osmani, Addy. *Learning JavaScript Design Patterns*. O'Reilly Media
2. Cohn, Stephen. *Angular in Action*. Manning Publications

3. Esposito, Dino. *Programming Microsoft ASP.NET 4*. Microsoft Press, 2011.

Online Resources:

1. **Microsoft Learn:** Official learning paths for ASP.NET Core, C#, and Azure. https://learn.microsoft.com/

2. **Angular Official Documentation** https://angular.io/docs**.**

Syllabus
Semester-VII

Course code: CAS41MEP401	**Course name:** Practical Based on React Java Script **Course category:** Major Elective
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20	
Pre-requisites: Knowledge of Object-Oriented Concepts, Database, HTML, CSS, JavaScript.	
Course Objectives: Project-based React JavaScript course is to equip learners with the skills to build and deploy dynamic, responsive single-page applications (SPAs) using modern front-end development practices.	
Course Outcomes: After completion of the course the student will be able to:	
CO1: Understand and implement the component-based architecture of React, breaking down complex interfaces into reusable, self-contained components.	
CO2: Implement conditional rendering and dynamically render lists of data using array.	
CO3: Study front-end technologies (HTML, CSS, and modern JavaScript ES6+) and apply them to build responsive UI layouts.	

List of Practicals

Practical No.	Content	Practical hours
1	Create a simple functional component that displays a "Hello World" message using JSX.	2
2	Demonstrate how to render dynamic content (e.g., current date/time, result of an arithmetic expression) within JSX using curly braces.	2
3	Compare the output of creating UI elements using pure JavaScript versus JSX.	2
4	Build an application with multiple nested components (parent-child relationship).	2
5	Program to Pass data from a parent to a child component using props.	2

6	Implement a counter application that increments, decrements, and resets a value, demonstrating how to use the use state hook to manage the component's state.	2
7	Create a form with various input elements (text, checkboxes, dropdowns) controlled by the React state.	2
8	Implement basic client-side form validation and handle form submission.	2
9	Implement handling user events like clicks and input changes, binding event handlers correctly.	2
10	Implement basic CRUD (Create, Read, Update, Delete) operations for tasks.	2
11	Write a program to include features like marking tasks as complete, filtering between active/completed tasks, and persisting data in local storage.	2
12	Develop a dynamic quiz that loads questions (from a local JSON or mock data source), tracks user answers, and calculates a final score.	2
13	Write a program to implement the use Effect hook to manage side effects and asynchronous data fetching.	2
14	Write a program to implement use React Router to handle multi-page navigation (home, individual posts, about page).	2
15	Case Study.	10

Textbooks:

1) Learning React: Modern Patterns for Developing React Apps by Alex Banks and Eve Porcello.

2) The Road to React: Your Journey to Master Plain yet Pragmatic React.js by Robin Weirich.

Handbooks:

1) Full stack React: The Complete Guide to ReactJS and Friends by Anthony Accomazzo et al

2) *Learning React: Functional Web Development with React and Redux* by Alex Banks & Eve Porcello.

Syllabus
Semester-VII

Course code: CAS41MEP402 (ASP.Net, Angular)	**Course Name**: Practical Based on Frontend Development **Course Category**: Major Elective	
Credits: 1	**Teaching scheme:** L-0 P-2	**Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Proficiency in **HTML, CSS, and JavaScript**.		
Course Objective:		
The main objective is to solve the customer's requirements.		
Course Outcome: After completion of the course the student will be able:		
CO1: Implement responsive web layouts and style interfaces using modern CSS techniques and frameworks.		
CO2: Develop Angular components, services, and routing for single-page applications.		
CO3: Consume RESTful APIs from an Angular application and perform CRUD operations.		
CO4: Set up a basic ASP.NET Core API to serve data to an Angular frontend.		
CO5: Deploy an integrated Angular and ASP.NET Core web application.		

Contents:

Practical No.	Description	Teaching Hours
1	- Set up Node.js, Angular CLI, and VS Code. - Create a basic Angular project.	2
2	- Create and understand a basic Angular component: template, styling, component logic. - Implement data binding (interpolation, property binding, event binding).	2
3	- Create and inject Angular services for shared logic. - Implement custom structural and attribute directives to manipulate the DOM.	2
4	- Configure Angular routing to create a multi-page SPA. - Create navigation links and handle route parameters.	2
5	- Implement both template-driven and reactive forms in Angular. - Apply form validation (built-in and custom validators).	2
6	- Set up a mock backend or use a public API. - Implement Angular's HttpClient module to perform GET, POST, PUT, DELETE operations.	2

7	- Create a new ASP.NET Core Web API project. - Configure the project to serve static Angular files.	2
8	- Create API controllers with actions for CRUD operations. - Implement model binding and data transfer objects.	2
9	- Implement a basic token-based authentication flow (e.g., JWT conceptual). - Implement Angular interceptors to attach authentication tokens to outgoing requests.	2
10	- Build the Angular application for production. - Deploy the integrated Angular and ASP.NET Core application to a local server	2

Books and References:

Text Books

1. Freeman, Adam. *Essential Angular for ASP.NET Core MVC*. Apress, 2017.

2. Freeman, Adam. *Pro ASP.NET Core MVC*. Apress, latest edition.

Reference Books:

1. Osmani, Addy. *Learning JavaScript Design Patterns*. O'Reilly Media
2. Cohn, Stephen. *Angular in Action*. Manning Publications

3. Esposito, Dino. *Programming Microsoft ASP.NET 4*. Microsoft Press, 2011.

Online Resources:

2. **Microsoft Learn:** Official learning paths for ASP.NET Core, C#, and Azure. https://learn.microsoft.com/

2. **Angular Official Documentation** https://angular.io/docs**.**

Syllabus
Semester VII

Course code: CAS41RML401 **Course Name**: Research Methodology **Course Category**: Research Methodology
Credits:3 **Teaching scheme:** L-**3** P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic understanding of subject fundamentals, elementary statistics, and academic reading and writing skills.
Objective: To develop an understanding of research methodology principles and techniques applicable to problem identification, analysis.
Course Outcome: After completion of the course the student will be able:
CO1: To develops the ability to identify research problems and formulate objectives and hypotheses.
CO2: To familiarizes students with research design, data collection, and analysis techniques relevant to robotics.
CO3: To enhances skills in technical writing, documentation, and research ethics.
CO4: Analyze and apply statistical techniques such as data processing, descriptive statistics, correlation, regression, and hypothesis testing to interpret and validate research data.
CO5: Evaluate research findings and create a structured research report or mini proposal using appropriate referencing styles, ethical practices, and computer-based research tools.

Unit	Contents	Teaching Hours.
1	**Introduction to Research &Research Process :**Meaning of Research, objectives, motivation, and significance of research, Types of research (basic/applied, qualitative/quantitative, conceptual/empirical) Research methods vs. research methodology, Research Approaches, Significance of Research, Scientific method and research ethics, Steps in the research process, Criteria of Good Research, Problems Encountered by Researchers in India	09
2	**Research Problem Formulation & Research Design:** Identification and definition of research problems in engineering/robotics (Meaning and sources of research problems, Criteria for selecting a research problem, Defining and formulating research problems), Review of literature and research gap identification, Formulation of objectives and hypotheses, Research design: exploratory, descriptive, experimental, Features of a good research design	09
3	**Sampling Design & Data Collection Methods:** Sampling concepts, population, sample, sampling errors , Probability and non-probability sampling techniques (Sample size (basic concept)Types of sampling errors (Sampling error, Non-sampling error, Bias and sources of bias), Methods of data collection: observation, interview, questionnaire, experiments, Selection of appropriate sampling and data collection methods (Nature of research problem, Objectives of the study, Time, cost, and resource constraints, Accuracy and reliability requirements, Relevance to robotics and engineering research)	09
4	**Data Processing, Analysis & Hypothesis Testing:** Data processing (Editing, coding, classification, tabulation), Descriptive statistics) Measures of central tendency, Measures of dispersion, Correlation and regression, Hypothesis testing (Concepts and Procedure, parametric and non-parametric tests), Interpretation of Statistical results.	09

5	**Research Reporting:** Interpretation of research results, Research report writing (Structure of research report technical paper, and project report), Referencing styles and bibliography, Plagiarism and ethical issues in research, Role of computers in research (Data analysis tools, Documentation and presentation tools), Preparation of a mini research proposal / report	09

Text Books :1.C.R. Kothari — Research Methodology: Methods & Techniques 2.Ranjit Kumar — Research Methodology: A Step-by-Step Guide for Beginners
Reference Books: 1.Garg, B.L., Karadia, R., Agarwal, R., & Agarwal, U.K. — An Introduction to Research Methodology 2.Paul D. Leedy& Jeanne Ellis Ormrod — Practical Research: Planning and Design
Online Resources: 1. Alison – Essentials of Research Methodology

Syllabus
Semester-VII

Course Code: CAS41RMP401 **Course Name:** Practical Based on Research Methodology **Course Category:** Research Methodology
Credits: 1 **Teaching Scheme:**L-0, P-**2** **Evaluation Scheme**:CA-30,ESE-20
Pre-requisites: Research Methodology practical is a basic understanding of research design, data collection methods, and fundamental statistical and analytical tools.
Course Objectives: The Research Methodology course is to equip students with the skills to design, conduct, analyze, and interpret research systematically and ethically.
Course Outcomes: At the end of the course, the students will be able to -
CO1: To develop the ability to formulate research problems and design appropriate research methodologies.
CO2: To design data collection and analysis procedures using suitable statistical and analytical techniques.
CO3: To understand the principles of ethical research and effectively interpret and present research findings.

List of Practicals:

Sr.No.	Title of the Experiment
1	Identify research papers from journals/conferences and prepare a structured literature review.
2	Select a domain in Computer fraternity and define a clear research problem with objectives.
3	Write research objectives and formulate null and alternative hypotheses.
4	Develop a suitable research design for a selected problem (exploratory/descriptive/experimental).
5	Identify population, sample size, sampling technique, and justify the choice.
6	Design a questionnaire for data collection related to a Computer fraternity problem.
7	Collect sample data using questionnaire/observation and enter data in spreadsheet/software.
8	Perform basic statistical analysis (mean, median, standard deviation, graphs).
9	Apply an appropriate statistical test (t-test / chi-square) and interpret results.
10	Prepare and present a mini research report standard research paper format.

Text Book: 1**. Practical Research: Planning and Design –** Paul D. Leedy & Jeanne Ellis Ormrod
Reference Book: 1. Research Methodology: Methods and Techniques – C.R. Kothari

Semester: Eight

Syllabus
Semester-VIII

Course code: CAS41MML404		**Course Name**: Machine Learning with Python **Course Category**: Major Mandatory
Credits:3	**Teaching scheme:** L-3 P-0	**Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic Concepts of machine learning and python programming.		
Course Objective: To provide students with fundamental knowledge and practical skills in applying machine learning techniques using Python for data analysis, modeling, and real-world problem solving.		
.		
Course Outcome: After completion of the course the student will be able:		
CO1. To introduce fundamental concepts of Machine Learning.		
CO2. To develop hands-on skills in Python for data analysis and modeling.		
CO3. To understand data preprocessing techniques.		
CO4. To apply supervised and unsupervised learning algorithms.		
CO5. To evaluate and improve machine learning models		

Contents:

Unit	Contents	Teaching Hours.
1	**Introduction to Machine Learning and Python Foundations** **Basics of Machine Learning** Definition of Artificial Intelligence, Machine Learning, and Data Science Need and Applications of Machine Learning Types of Machine Learning Supervised Learning Unsupervised Learning Semi-supervised Learning Machine Learning Workflow Real-world examples of ML systems Overview of Python ecosystem for ML **Python for Data Handling :Introduction to NumPy** Array creation Indexing and slicing Shape and reshaping Basic mathematical operations **Introduction to Pandas** Creating DataFrames Reading CSV files Selecting rows and columns Filtering data Working with Datasets Understanding features and labels	**09**

	Basic data visualization Simple dataset exploration	
2	**Data Preprocessing and Data Preparation** **Data Understanding** Types of Data Structured and Unstructured Data Identifying missing values and outliers **Data Cleaning** Handling missing data Removal , Mean / Median imputation Removing duplicate records, Noise handling **Data Transformation** Scaling and Normalization Standardization Encoding categorical variables Label Encoding One-Hot Encoding Binning and Discretization	**09**
3	**Feature Engineering (Introductory)** Feature creation Polynomial features (basic idea) Feature interaction **Dataset Splitting** Training and Testing data Avoiding data leakage **Supervised Learning** Labeled datasets Input features and target variables Generalization concept Overfitting and Underfitting Relationship between Model Complexity and Dataset Size	**09**
4	**Classification Approach Concepts** Binary and Multiclass Classification Decision boundary Confusion matrix **Algorithms (Introductory)** k-Nearest Neighbors (k-NN) Naive Bayes Classifier Logistic Regression Decision Trees Support Vector Machines (Concept only) **Regression Approach Concepts** Continuous target prediction Simple and Multiple Linear Regression Relationship between variables	**09**

	Algorithms (Introductory) Linear Regression k-Nearest Neighbors Regression Decision Tree Regression Support Vector Regression (Concept only)	
5	**Unsupervised Learning and Feature Engineering** **Unsupervised Learning** Types of Unsupervised Learning Challenges in Unsupervised Learning **Dimensionality Reduction** Feature extraction Principal Component Analysis (PCA) Non-Negative Matrix Factorization (Concept) t-SNE for visualization **Clustering** k-Means Clustering Agglomerative Clustering DBSCAN Comparison of clustering techniques	**09**

Books and References:

Text Books 1. Practical Machine Learning for Data Analysis Using Python Abdulhamit Subasi. 2. Advance Machine Learning with Python by John Hearty.
Reference Books: 1. "Python Machine Learning: Machine Learning and Deep Learning with Python, scikitlearn, and Tensor Flow " by Sebastian Raschka and Vahid Mirjalili. 2. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron. 3. "Introduction to Machine Learning with Python".
Online Resources: **1.** https://youtu.be/7eh4d6sabA0?si=JWHNJRVS6NhQOPYe **2.** https://youtu.be/kqtD5dpn9C8?si=LBep4HWaMFRrAPsD **3.** https://youtu.be/4SJ7bEILPJk?si=5LurvjzUOuCew1W9

Syllabus
Semester-VIII

Course code: CAS41MML405 **Course Name**: Blockchain Platforms **Course Category**: Major Mandatory
Credits:3 **Teaching scheme:** L-3 P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Fundamental knowledge of Blockchain Technology, cryptography, Bitcoin, Ethereum, and smart contracts.
Course Objective: The objective of this course is to provide students with an in-depth understanding of major blockchain platforms, their architectures, tools, and real-world deployment models.
Course Outcome: After completion of the course the student will be able:
CO1. Understand the architecture and working of major blockchain platforms.
CO2. Compare public, private, and enterprise blockchain platforms.
CO3. Analyze Ethereum-based and non-Ethereum blockchain ecosystems.
CO4. Understand blockchain-as-a-service platforms and deployment models.
CO5. Identify suitable blockchain platforms for real-world and enterprise use cases.

Contents:

Unit	Contents	Teaching Hours.
1	**Overview of Blockchain Platforms:** Concept of blockchain platforms, classification of platforms, public vs private vs consortium platforms, criteria for selecting a blockchain platform, platform architecture overview, comparison of popular blockchain platforms.	**09**
2	**Ethereum and Related Platforms:** Ethereum platform architecture, Ethereum Virtual Machine, smart contract execution model, Ethereum development stack, Polygon (Layer 2) overview, Binance Smart Chain overview, advantages and limitations of Ethereum-based platforms.	**09**
3	**Enterprise Blockchain Platforms:** Hyperledger ecosystem overview, Hyperledger Fabric architecture, peers, orderers, channels, chaincode concept, Hyperledger Sawtooth overview, enterprise use cases of permissioned blockchains.	**09**
4	**Other Blockchain Platforms:** Introduction to Ripple and Stellar platforms, Corda platform architecture, comparison of Corda with traditional blockchain systems, use cases in banking and financial services, interoperability concepts.	**09**
5	**Blockchain Deployment and Services:** Blockchain-as-a-Service (BaaS) overview, cloud blockchain platforms (AWS, Azure, IBM Blockchain), deployment challenges, scalability and security considerations, governance models, future trends in blockchain platforms.	**09**

Books and References:

Text Books **1.** Imran Bashir, *Mastering Blockchain*, Packt Publishing. **2.** Chandramouli Subramanian et al., *Blockchain Technology*, Universities Press.
Reference Books: **1.** Andreas M. Antonopoulos, *Mastering Bitcoin* and *Mastering Ethereum*, O'Reilly. **2.** Daniel Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*. **3.** Ritesh Modi, *Solidity Programming Essentials*.
Online Resources: **1.** Ethereum Documentation – https://ethereum.org/developers/ **2.** Hyperledger Documentation – https://www.hyperledger.org/use **3** Corda Documentation – https://docs.r3.com/ **4.** NPTEL – Blockchain Architecture Design and Use Cases (IITs)

Syllabus
Semester-VIII

Course code: CAS41MML406		**Course Name**: Biomedical Image Processing **Course Category**: Major Mandatory
Credits:3	**Teaching scheme:** L-3 P-0	**Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic knowledge of signals and systems, linear algebra, probability and statistics, and programming fundamentals (MATLAB/Python), with an introductory understanding of human anatomy or medical imaging concepts preferred.		
Course Objective: To provide fundamental knowledge of digital image processing and biomedical image preprocessing and segmentation techniques, while developing practical problem-solving skills through the application of algorithms to medical imaging problems.		
Course Outcome: After completion of the course the student will be able:		
CO1. Ability to understand the various medical images and their difference.		
CO2. Ability to learn different image enhancement techniques used for medical images.		
CO3. Ability to learn various medical image segmentation techniques.		
CO4. Ability to study the various feature extraction techniques and classifier models for biomedical image.		
C05. Ability classifier models for biomedical image.		

Contents:

Unit	Contents	Teaching Hours.
1	**Fundamentals of Image processing and Image Transforms** **Basic steps of Image processing system** **Sampling and Quantization of an Image** **Basic relationship between Pixels** **Image Transforms:** 2 – D Discrete Fourier Transform Discrete Cosine Transform (DCT) **Discrete Wavelet transforms** **Various Medical images:** X-ray and Computed Tomography (CT) imaging Magnetic Resonance Imaging (MRI) Ultrasonic Imaging, Microscopic Imaging.	**09**
2	**Image Enhancement techniques:** Gray scale thresholding Contrast manipulation Histogram equalization Laplacian derivatives **Rank operators – textural analysis** **Homomorphic filtering** **Advanced Enhancement Techniques**	**09**

	Multi-scale enhancement Wavelet-based image enhancement, AI-assisted image enhancement, Clinical relevance and ethical considerations	
3	**Introduction to Image Segmentation** Definition and objectives of image segmentation Importance of segmentation in biomedical applications Challenges in medical image segmentation (noise, intensity overlap, artifacts) **Edge Detection Techniques** Concept of edges in biomedical images Gradient-based edge detection Second-order edge detection Canny edge detector (optimal edge detection criteria) Applications in tumor boundary and organ detection **Thresholding-Based Segmentation** Principle of thresholding Global and local thresholding Optimal thresholding Otsu's thresholding method Applications in cell counting and tissue segmentation **Region, Clustering, and Soft Computing-Based Segmentation** Region-Based Segmentation Clustering-Based Segmentation Fuzzy-Based Segmentation Techniques	**09**
4	**Shape analysis in biomedical imaging: significance and applications** **Shape and contour representation techniques** Boundary-based representation Region-based representation Contour extraction methods **Shape factors and morphological descriptors** Geometric features (area, perimeter, centroid, axes) Shape factors: aspect ratio, circularity, compactness, solidity, convexity, elongation Invariant shape features (translation, rotation, scale invariance) **Fourier Descriptors for shape representation** Contour parameterization Discrete Fourier Transform (DFT) of shapes Properties of Fourier descriptors Shape reconstruction and noise sensitivity	**09**
5	**Challenges in Biomedical Image Acquisition and Analysis** Noise, artifacts, variability in imaging modalities, resolution issues **Feature Extraction for Medical Images** Shape-based, texture-based, and intensity-based features	**09**

	Feature Selection Techniques Dimensionality reduction, relevance and redundancy, impact on classification performance **Classifier Models for Medical Applications** • Support Vector Machines (SVM) • Artificial Neural Networks (ANN) • Naïve Bayes classifier • k-Nearest Neighbor (k-NN) classifier	

Books and References:

Text Books 1. Gonzalez R. C. and Woods R. E, "Digital Image Processing", Pearson Prentice Hall, 2nd edition, 2002. 2. Rangaraj M. Rangayyan, “Biomedical Image Analysis”, CRC Press, 2000. 3. Qiang Wu, Fatima A. Merchant, Kenneth R. Castleman, “Microscope Image Processing”, Elsevier Publication, ISBN: 978-0-12-372578-3.
Reference Books: 1. Gonzalez R. C, Woods R. E and Eddins S. L "Digital Image Processing using MATLAB", McGraw Hill Education, 2nd edition, 2017. 2. Richard O. Duda, Peter E. Hart, David G. Stork. Pattern classification, Wiley, New York, 2001.
Online Resources: **4.** https://www.nitrc.org/projects/cbica_tutorials/ **5.** https://medium.com/@protobioengineering/a-dozen-resources-for-learning-how-to-analyze-medical-images-with-code-38f3926484ab **6.** https://onlinecourses.nptel.ac.in/noc22_bt56/preview

Syllabus
Semester-VIII

Course code: CAS41MMP404 **Course Name**: **Practical Based on** Machine Learning with Python **Course Category**: Major Mandatory
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic Concepts of machine learning and python programming.
Course Objective: To provide students with fundamental knowledge and practical skills in applying machine learning techniques using Python for data analysis, modeling, and real-world problem solving.
Course Outcome: After completion of the course the student will be able:
CO1. To introduce fundamental concepts of Machine Learning.
CO2. To develop hands-on skills in Python for data analysis and modeling.
CO3. To understand data preprocessing techniques.
CO4. To apply supervised and unsupervised learning algorithms.
CO5. To evaluate and improve machine learning models

List of Practical

Practical No.	Contents	Practical Hours.
1	Install and configure the Python environment for machine learning using Anaconda and Jupyter Notebook. Write a program in Python for performing basic NumPy operations.	**2**
2	Write a program in Python for data handling and manipulation using Pandas.	**2**
3	Write a program in Python for exploratory data analysis and data visualization.	**2**
4	Write a program in Python for identifying and handling missing values in a dataset.	**2**
5	Write a program in Python for data transformation and categorical feature encoding.	**2**
6	Write a program in Python for feature engineering and splitting the dataset into training and testing sets.	**2**
7	Write a program in Python to implement k-Nearest Neighbors and Naive Bayes classification.	**2**
8	Write a program in Python to perform classification using Logistic Regression and Decision Tree.	**2**
9	Write a program in Python to implement regression techniques using machine learning.	**2**

10	Write a program in Python for clustering and dimensionality reduction using PCA.	**2**
11	Project	**10**

Books and References:

Text Books 1. Practical Machine Learning for Data Analysis Using Python Abdulhamit Subasi. 2. Advance Machine Learning with Python by John Hearty.
Reference Books: 1. "Python Machine Learning: Machine Learning and Deep Learning with Python, scikitlearn, and Tensor Flow " by Sebastian Raschka and Vahid Mirjalili. 2. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron. 3. "Introduction to Machine Learning with Python".
Online Resources: **1.** https://youtu.be/7eh4d6sabA0?si=JWHNJRVS6NhQOPYe **2.** https://youtu.be/kqtD5dpn9C8?si=LBep4HWaMFRrAPsD **3.** https://youtu.be/4SJ7bEILPJk?si=5LurvjzUOuCew1W9

Syllabus
Semester-VIII

Course code: CAS41MMP405 **Course Name**: Practical based on Blockchain Platform **Course Category**: Major Mandatory
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of computers, programming fundamentals, and internet usage.
Course Objective: The objective of this practical course is to provide hands-on understanding of different blockchain platforms, enterprise blockchain frameworks, and blockchain deployment models.
Course Outcome: After completion of the course the student will be able:
CO1. Identify and compare different blockchain platforms and architectures.
CO2. Work with Ethereum-based and enterprise blockchain platforms.
CO3. Analyze permissioned blockchain frameworks used in industry.
CO4. Understand blockchain deployment and cloud-based services.
CO5. Select appropriate blockchain platforms for real-world use cases.

Contents:

Sr. no.	Contents
1	Install and configure MetaMask wallet and connect to an Ethereum test network.
2	Deploy a simple smart contract on Ethereum test network and observe transaction details.
3	Create and deploy an ERC-20 token using OpenZeppelin contracts on a test network.
4	Write a code to Interact with a deployed smart contract using Remix and MetaMask.
5	Setup Ganache local blockchain and deploy a smart contract on a local network.
6	Install and configure Truffle framework and migrate a smart contract to Ganache.
7	Develop a simple DApp frontend to interact with smart contract using Web3.js.
8	Deploy and test a smart contract using Ethereum test network and analyze gas usage.
9	Setup a basic Hyperledger Fabric test network using sample scripts (study + execution).
10	Case study implementation: Select a blockchain platform and justify its use for a given application.
11	project

Books and References:

Text Books

1. Imran Bashir, *Mastering Blockchain*, Packt Publishing.
2. Chandramouli Subramanian et al., *Blockchain Technology*, Universities Press.

Reference Books: **1.** Andreas M. Antonopoulos, *Mastering Bitcoin* and *Mastering Ethereum*, O'Reilly. **2.** Daniel Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*. **3.** Ritesh Modi, *Solidity Programming Essentials*.
Online Resources: **1.** Ethereum Documentation – https://ethereum.org/developers/ **2.** Corda Documentation – https://docs.r3.com/ **3.** Hyperledger Documentation – https://www.hyperledger.org/use **4.** NPTEL – Blockchain Architecture Design and Use Cases (IITs)

Syllabus
Semester-VIII

Course code: CAS41MMP406 **Course Name**: Practical Based on Biomedical Image Processing **Course Category**: Major Mandatory
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of signals and systems, linear algebra, probability and statistics, and programming fundamentals (MATLAB/Python), with an introductory understanding of human anatomy or medical imaging concepts preferred.
Course Objective: To provide fundamental knowledge of digital image processing and biomedical image preprocessing and segmentation techniques, while developing practical problem-solving skills through the application of algorithms to medical imaging problems.
Course Outcome: After completion of the course the student will be able:
CO1. Ability to understand the various medical images and their difference.
CO2. Ability to learn different image enhancement techniques used for medical images.
CO3. Ability to learn various medical image segmentation techniques.
CO4. Ability to study the various feature extraction techniques
C05. Ability classifier models for biomedical image.

Contents:

Practical No.	Contents	Practical Hours.
1	Program to study the biomedical image processing environment and perform basic image input–output operations, including reading, displaying, and analyzing medical images (DICOM) and their properties.	**2**
2	Program to read, display, and analyze biomedical images and perform grayscale and intensity transformations.	**2**
3	Program to enhance biomedical images using contrast enhancement and histogram equalization techniques.	**2**
4	Program to simulate noise in medical images and apply spatial and frequency domain filtering for noise removal.	**2**
5	Program to implement morphological operations such as erosion, dilation, opening, and closing on biomedical images.	**2**
6	Program to perform edge detection and boundary extraction using Sobel, Prewitt, and Canny operators.	**2**
7	Program to implement threshold-based segmentation techniques including global, adaptive, and Otsu's methods.	**2**

8	Program to perform region-based and clustering-based segmentation using region growing, watershed, and k-means algorithms.	**2**
9	Program to extract features from segmented biomedical images including shape, texture, and intensity features.	**2**
10	Program to classify medical images using machine learning classifiers such as SVM, k-NN, Naïve Bayes, and ANN.	**2**
11	Project	**10**

Books and References:

Text Books **1.** Imran Bashir, *Mastering Blockchain*, Packt Publishing. **2.** Chandramouli Subramanian et al., *Blockchain Technology*, Universities Press.
Reference Books: **1.** Andreas M. Antonopoulos, *Mastering Bitcoin* and *Mastering Ethereum*, O'Reilly. **2.** Daniel Drescher, *Blockchain Basics: A Non-Technical Introduction in 25 Steps*. **3.** Ritesh Modi, *Solidity Programming Essentials*.
Online Resources: **1.** Ethereum Documentation – https://ethereum.org/developers/ **2.** Corda Documentation – https://docs.r3.com/ **3.** Hyperledger Documentation – https://www.hyperledger.org/use **4.** NPTEL – Blockchain Architecture Design and Use Cases (IITs)

Syllabus
Semester-VIII

Course code: CAS41MEL403 **Course Name**: Grid & Cloud Computing **Course Category**: Major Elective
Credits:3 **Teaching scheme:** L-3 P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic knowledge of computers, operating systems, computer networks, and internet technologies.
Course Objective: The objective of this course is to introduce students to the concepts, architecture, models, and applications of grid and cloud computing, enabling them to understand modern distributed computing environments.
Course Outcome: After completion of the course the student will be able:
CO1. Understand the fundamentals and architecture of grid and cloud computing.
CO2. Explain virtualization, service models, and deployment models in cloud computing.
CO3. Analyze cloud infrastructure, resource management, and scheduling techniques.
CO4. Identify cloud security issues and solutions.
CO5. Understand real-world applications and emerging trends in grid and cloud computing.

Contents:

Unit	Contents	Teaching Hours.
1	**Introduction to Grid Computing:** Distributed computing concepts, evolution of grid computing, grid architecture and components, grid middleware, resource sharing and coordination, applications of grid computing, advantages and limitations of grid computing.	**09**
2	**Cloud Computing Fundamentals:** Introduction to cloud computing, characteristics of cloud computing, cloud computing architecture, service models – IaaS, PaaS, SaaS, deployment models – public, private, hybrid and community cloud, comparison of grid and cloud computing.	**09**
3	**Virtualization and Cloud Infrastructure:** Concept of virtualization, types of virtualization, hypervisors, virtual machines, cloud infrastructure components, data centers, resource provisioning, load balancing, cloud storage models.	**09**
4	**Cloud Security and Management:** Cloud security issues and challenges, data security and privacy, identity and access management, service level agreements (SLA), cloud monitoring, metering and billing, fault tolerance and disaster recovery.	**09**
5	**Cloud Platforms and Applications:** Overview of cloud platforms such as AWS, Microsoft Azure and Google Cloud (conceptual), cloud-based applications, cloud computing use cases in education, healthcare, banking and e-commerce, limitations of cloud computing, future trends and scope.	**09**

Books and References:

Text Books **1.** Buyya R., Broberg J., Goscinski A., *Cloud Computing: Principles and Paradigms*, Wiley. **2.** Ahmar Abbas, *Grid Computing: A Practical Guide to Technology and Applications*, Firewall Media.
Reference Books: **1.** Rajkumar Buyya, *Mastering Cloud Computing*, McGraw-Hill. **2.** Anthony T. Velte, Toby J. Velte, *Cloud Computing – A Practical Approach*, McGraw-Hill. **3.** Sosinsky B., *Cloud Computing Bible*, Wiley.
Online Resources: **1.** NPTEL – Cloud Computing (IIT Kharagpur / IIT Madras) **2.** AWS Cloud Computing Overview – https://aws.amazon.com **3** Microsoft Azure Documentation – https://learn.microsoft.com/azure **4.** Google Cloud Documentation – https://cloud.google.com/docs

Semester VIII

Course code: CAS41MEL404 **Course Name**: Big Data Analytics **Course Category**: Major Elective
Credits: 3 **Teaching Scheme:** L-3 P-0 **Evaluation Scheme:** CA–60, ESE–40
Pre-requisites: Proficiency in a programming language like Java or Python
Course Objective:
To enable students to apply various analytical techniques and tools for processing and deriving insights from large datasets.
Course Outcome: After completion of the course the student will be able:
CO1. Analyze the characteristics and challenges of Big Data and its impact on various domains.
CO2. Implement solutions for Big Data storage and processing using frameworks such as Hadoop and Spark.
CO3. Apply appropriate Big Data analytic techniques to extract meaningful insights from large datasets.
CO4: Utilize appropriate Big Data analytic techniques and tools to extract meaningful insights from large datasets.
CO5: Evaluate Big Data case studies, propose solutions for complex analytical problems, and consider ethical implications.

Contents:

Unit	Contents	Teaching Hours.
1	**Introduction to Big Data and its Ecosystem:** - Understanding Big Data: Characteristics, challenges, and opportunities. - Big Data Architectures: Lambda and Kappa architectures. - Introduction to Hadoop: HDFS architecture, MapReduce programming model. - Overview of the Hadoop Ecosystem: YARN, Hive, Pig, Sqoop.	8
2	**NoSQL Databases** - Introduction to NoSQL: Motivations, CAP theorem, types of NoSQL databases. - Document Databases: MongoDB (data model, basic operations). - Column-Family Databases: Cassandra (data model, basic operations) - Comparison of NoSQL databases with traditional RDBMS.	9
3	**Data Ingestion and Warehousing** : - Data Ingestion: Batch vs. Stream processing, data connectors, real-time data ingestion. - Data Warehousing concepts in a Big Data context: Data Lake vs. Data Warehouse. - Data Governance and Metadata Management in Big Data environments. - ETL vs. ELT in Big Data.	9

4	**Big Data Processing with Spark** : - Introduction to Apache Spark: RDDs, Spark Architecture, comparison with MapReduce. - Spark Core: Transformations and Actions, lazy evaluation. - Spark SQL: Working with DataFrames and Datasets, Catalyst Optimizer. - Spark Streaming/Structured Streaming: Real-time data processing concepts. - Introduction to PySpark/Spark Scala.	10
5	**Big Data Analytics Techniques and Applications** - Data Preprocessing and Cleaning for Big Data. - Machine Learning with Big Data: Overview of MLlib, common algorithms (clustering, classification). - Big Data Visualization: Tools and techniques for presenting insights - Ethical considerations and privacy in Big Data.	9

Books and References:

Text Books **1.** Kumar, J., Kumar, A., & Kumar, R.. *Big Data and Analytics: The key concepts and practical applications of big data analytics*. BPB Publications **2.** Demchenko, Y., Cuadrado-Gallego, J. J., Chertov, O., & Nordström, L.. *Big Data Infrastructure Technologies for Data Analytics*
Reference Books: **1.** White, T.. *Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale* (4th ed.). O'Reilly Media **2.** Rajaraman, A., & Ullman, J. D.. *Mining of Massive Datasets*. Cambridge University Press
Online Resources: **1.** Official documentation for Apache Hadoop, Apache Spark, MongoDB, Cassandra, etc.

Syllabus
Semester-VIII

Course code: CAS41MEP403 **Course Name**: Practical based on Grid & Cloud Computing **Course Category**: Major Elective
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of computers, operating systems, computer networks, and internet usage.
Course Objective: The objective of this practical course is to provide hands-on exposure to grid and cloud computing concepts, virtualization, cloud services, and real-world cloud applications.
Course Outcome: After completion of the course the student will be able:
CO1. Understand the architecture and working of grid and cloud computing systems.
CO2. Work with virtualization and cloud service models.
CO3. Use basic cloud services such as compute and storage.
CO4. Identify cloud security, monitoring, and management aspects.
CO5. Analyze real-world applications of grid and cloud computing.

Contents:

Sr. no.	Contents
1	Study of Grid Computing architecture, components, middleware, and applications.
2	Comparison of Grid Computing and Cloud Computing with suitable diagrams.
3	Create and use a virtual machine using free virtualization software.
4	Study of cloud service models (IaaS, PaaS, SaaS) and deployment models.
5	Create and configure a free cloud account and explore basic cloud services.
6	Launch and manage a virtual machine instance on a cloud platform.
7	Implement and use cloud storage services for data upload and management.
8	Implement cloud security concepts including identity and access management.
9	Implement of cloud monitoring, billing, and cost management features.
10	Case study on cloud computing applications in real-world domains.
11	Project

Books and References:

Text Books

1. Buyya R., Broberg J., Goscinski A., *Cloud Computing: Principles and Paradigms*, Wiley.
2. Ahmar Abbas, *Grid Computing: A Practical Guide to Technology and Applications*,

Firewall Media.
Reference Books: **1.** Rajkumar Buyya, *Mastering Cloud Computing*, McGraw-Hill. **2.** Anthony T. Velte, Toby J. Velte, *Cloud Computing – A Practical Approach*, McGraw-Hill. **3.** Sosinsky B., *Cloud Computing Bible*, Wiley.
Online Resources: **1.** NPTEL – Cloud Computing (IITs) **2.** AWS Cloud Documentation – https://aws.amazon.com **3.** Microsoft Azure Documentation – https://learn.microsoft.com/azure **4.** Google Cloud Documentation – https://cloud.google.com/docs

Syllabus
Semester-VIII

Course code: CAS41MEP404 **Course Name**: Practical Based on Big Data Analytics **Course Category**: ME
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Proficiency in HTML, CSS, and JavaScript.
Course Objective:
The main objective is to solve the customer's requirements.
Course Outcome: After completion of the course the student will be able:
CO1: Set up and interact with a Big Data environment, including Hadoop HDFS and basic MapReduce jobs.
CO2: Implement data storage and retrieval operations using NoSQL databases like MongoDB and Cassandra.
CO3: Write and execute data processing scripts using Apache Spark for various data manipulation tasks.
CO4: Perform basic data ingestion and preprocessing steps on large datasets.
CO5: Apply fundamental Big Data analytical techniques and visualize results using appropriate tools.

Contents:

Practical No.	Description	Teaching Hours
1	Set up a single-node Hadoop cluster, Perform basic HDFS commands: put, get, ls, mkdir, cat.	2
2	Write a basic MapReduce program (e.g., Word Count) in Java or Python	2
3	Compile and execute the MapReduce job on the Hadoop cluster., Analyze the output files generated by the MapReduce job	2
4	Install and configure MongoDB. Create a database and collections.	2
5	Insert, find, update, and delete documents using MongoDB shell commands or a Python client.	2
6	Set up a single-node Cassandra instance. Create keyspaces and tables.	2
7	Insert data, query data using CQL. Understand the basic differences between MongoDB and Cassandra in practical.	2
8	Set up Spark in local mode. Load data (e.g., from a text file) into an RDD.	2

9	Perform basic RDD transformations (map, filter, flatMap) and actions (count, collect).	2
10	Implement a simple data cleaning task using RDDs.	2
11	Mini Project / Case Study	10

Books and References:

Text Books

1. Kumar, J., Kumar, A., & Kumar, R.. *Big Data and Analytics: The key concepts and practical applications of big data analytics*. BPB Publications

2. Demchenko, Y., Cuadrado-Gallego, J. J., Chertov, O., & Nordström, L.. *Big Data Infrastructure Technologies for Data Analytics*

Reference Books:

1. White, T.. *Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale* (4th ed.). O'Reilly Media

2. Rajaraman, A., & Ullman, J. D.. *Mining of Massive Datasets*. Cambridge University Press

Online Resources:

1. Official documentation for Apache Hadoop, Apache Spark, MongoDB, Cassandra, etc.

B.C.A.(Hons. With Research)
Semester: SEVEN

Syllabus
Semester-VII

Course Code: CAS41MML407 **Course Name:** Geospatial Information System **Course Category:** Major Mandatory
Credits: 3 **Teaching Scheme:**L-3 P-0 **Evaluation Scheme:** CA-60, ESE-40
Pre-requisites: Fundamental knowledge of geography, maps, and basic computer applications
Course Objectives: To understand GIS fundamentals, spatial data modelling, data management, data acquisition and accuracy, along with image processing, applications, and modern GIS technologies**.**
Course Outcomes: At the end of the course, the students will be able to -
CO1: Explain fundamentals, components, and operations of GIS.
CO2: Describe spatial data models, data sources, and data input methods.
CO3: Understand GIS database management systems and data organization.
CO4: Apply data acquisition methods, accuracy assessment, and geodatabase management.
CO5:Explain image processing techniques, GIS applications, and modern GIS technologies

Course Contents –

Unit	Content	Teaching hours
1	**Fundamentals of GIS** Introduction**,** Roots of GIS**,** Overview of Information System**,** Contribution Disciplines**,** GIS Definitions and Terminology**,** Components of GIS, GIS Queries**,** GIS Architecture**,** Theoretical Models of GIS**,** Theoretical Framework for GIS**,** GIS Categories**,** Levels/Scales of Measurement, Function of GIS and GIS Operations	5
2	**Representation and Modelling of Spatial Data** Introduction, Stages of GIS Data Modelling, Graphic Representation of Spatial Data, Raster GIS Models, Vector GIS Models, Comparison of Raster and Vector Models, Data sources: maps, GPS, satellite imagery, surveys, Data input methods and data editing	10
3	**GIS Data Management and Database Concepts** Introduction, Database Management Systems, GIS Data File Management, Database Models, Storage of GIS Data, Object-Based Data Models, Organizational Strategy of DBMS in GIS, Concepts of DBMS in Context of GIS, Hierarchical, Network, and Relational Models	10
4	**Data Acquisition, Accuracy, and Geodatabase Management**	10

	Data acquisition through scanners and digitizers, Methods of digitization, Geometric transformations of raster and vector data and transformation coefficients, Root Mean Square (RMS) error, Spatial data accuracy and sources of errors, Difference between accuracy and precision. **Geodatabase Management:** Basics of geodatabase model, Role of databases in GIS, Creating, editing, and managing geodatabases, Topology building and editing, Map composition and layout design	
5	**Image Processing, Applications and Modern GIS Technologies** Introduction to image processing in GIS, Digital image processing concepts, Image preprocessing and enhancement techniques, Image transformations and classification, Integration of image processing with GIS, GIS applications**, Modern GIS Technologies:** Web GIS and Mobile GIS, Integration with Remote Sensing and GPS, Future trends and emerging technologies in GIS	10

Text Books: 1. Essentials of Geographic Information Systems,Joseph K. Berry,Open Textbook Library (Free Online) 2.Fundamentals of Geographic Information System,Debashis Chakraborty & Rabi N. Sahoo,Viva Books, ISBN: 978-8130900414 3.Introduction to Geographic Information Systems,Falguni MukherjeeBloomsbury Publishing, ISBN: 979-8881856854
Reference Books: 1. Remote Sensing and GIS – Basudeb Bhatta – Oxford University Press India 2.Geographical Information Systems – O'Reilly Media 3.Open GIS – Jizhe Xia, Qunying Huang, Zhipeng Gui, Wei Tu – Springer Cham 4.GIS & Remote Sensing: Recent Trends and Applications – Udgata, Kumar, Chavan, et al. – KD Publications
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus
Semester VII

Course code: CAS41MML408 **Course Name**: Signal Processing **Course Category**: Major Mandatory
Credits:3 **Teaching scheme:** L-3 P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic knowledge of Basic mathematics, including calculus, complex numbers, and elementary linear algebra, is required for this course.
Course Objective: To provide students with a strong foundation in signal processing concepts, including signal representation, system analysis, time- and frequency-domain techniques, and their applications in modern engineering systems.
.
Course Outcome: After completion of the course the student will be able:
CO1. Introduce fundamental concepts of signal processing, including signal classification and system properties.
CO2. To analyze signals using time-domain and frequency-domain techniques such as convolution and Fourier transforms.
CO3. To apply sampling and basic digital signal processing concepts to real-world signals.
CO4. To design and implement basic signal processing solutions using computational tools for engineering applications.
CO5. To implement basic signal processing solutions using computational tools for engineering applications.

Contents:

Unit	Contents	Teaching Hours.
1	**Introduction to Signal Processing and Signals** **Fundamentals of Signal Processing and Signals** Meaning and scope of Signal Processing Evolution of Signal Processing Analog Signal Processing and Digital Signal Processing Role of Signal Processing in modern engineering Applications of Signal Processing in communication systems, biomedical engineering, multimedia, Internet of Things (IoT), and Artificial Intelligence (AI). Definition of a signal Energy and Power signals Periodic and Aperiodic signals	**5**
2	**Mathematical representation of signals** Analytical representation Graphical representation Common signal notations	**10**

	Classification of signals Continuous-time and Discrete-time signals Deterministic and Random signals	
3	**Signal Operations and System Concepts** **Basic signal operations** Time shifting Time scaling Time reversal Amplitude scaling **Concept of a system** **Classification of systems** Linear and Nonlinear systems Time-invariant and Time-variant systems Causal and Non-causal systems Stable and Unstable systems	**10**
4	**Time-Domain Analysis of Signals and Systems** Elementary signals Unit impulse Unit step Ramp Exponential, and sinusoidal signals Impulse response and system characterization Continuous-time and Discrete-time convolution Properties of convolution Interconnection of systems Introduction to correlation and its applications	**10**
	Frequency-Domain Analysis Introduction for frequency-domain representation Fourier Series (trigonometric and exponential forms) Continuous-Time Fourier Transform (CTFT) Discrete-Time Fourier Transform (DTFT) Discrete Fourier Transform (DFT) Fast Fourier Transform (FFT) Spectral analysis and practical interpretation	

Books and References:

Text Books

1. **Signals and Systems** by Oppenheim & Willsky: The classic for understanding the core concepts of signals and systems before diving deep into digital.
2. **Understanding Digital Signal Processing** by Richard G. Lyons: Praised for its

clarity and practical approach, great for beginners.

Reference Books:

1. **Understanding Digital Signal Processing** by Richard G. Lyons: Highly recommended for grasping DSP fundamentals intuitively.
2. **The Scientist & Engineer's Guide to Digital Signal Processing** by Steven W. Smith: Excellent for beginners, available online.

Online Resources:

7. https://github.com/openlists/DSPResources
8. https://www.mathworks.com/academia/courseware/teaching-signal-processing-and-communications-with-matlab-and-simulink.html

Syllabus

Semester-VII

Course Code: CAS41MMP406 **Course Name:** Practical Based on Geospatial Information System **Course Category:** Major Mandatory
Credits: 1 **Teaching Scheme:**L-0,P-2 **Evaluation Scheme**:CA-30, ESE-20
Pre-requisites: Basic knowledge of computers and introductory GIS concepts.
Course Objectives: To provide practical training in GIS software, focusing on QGIS for spatial data handling, analysis, and modern GIS applications
Course Outcomes: At the end of the course, the students will be able to -
CO1: Explain GIS software such as QGIS, ArcGIS, ERDAS, and ENVI and explore the QGIS interface.
CO2: Add, visualize, and manage raster and vector data layers using QGIS.
CO3: Create, edit, label, and query vector data using QGIS tools.
CO4: Apply map projections, reprojection, measurements, and georeferencing techniques in QGIS.
CO5: Analyze raster datasets and understand modern GIS applications including Web GIS and Mobile GIS.

List of Practicals:

Sr.No.	Title of the Experiment	Practical Hours
1	Explain QGIS, ArcGIS, ERDAS, ENVI software; installation and explore QGIS interface, menu bar, and toolbar	2
2	Insert raster and vector layers in QGIS and study the properties of data layers.	2
3	Change visualization of raster and vector data using symbology options in QGIS.	2
4	Create vector layers (point, line, polygon) using new shapefile layer in QGIS.	2
5	Label vector data using QGIS label properties.	2
6	Study map projections and perform reprojection of raster and vector layers in QGIS.	2
7	Perform feature selection and attribute querying using QGIS query tools.	2
8	Edit vector data using toggle editing and overlay different vector layers in QGIS.	2
9	Measure area, distance, and angles using QGIS measure tools.	2
10	Perform georeferencing of raster images using the Georeferencer tool in QGIS and Visualize and analyze raster datasets; understand modern GIS applications (Web GIS, Mobile GIS concepts).	2
11	Project	10

Reference Book / Hand Books/ Lab Manual

1.Burrough, P.A. and McDonnell, R.A., *Principles of Geographical Information Systems*, Oxford University Press.
2. QGIS Documentation and Training Manual (Open Source Geospatial Foundation – OSGeo).
3. Longley, P.A., Goodchild, M.F., Maguire, D.J., and Rhind, D.W., *Geographic Information Systems and Science*, Wiley.

Syllabus
Semester VII

Course code: CAS41MMP407 **Course Name**: Practical based on Signal Processing **Course Category**: Major Mandatory
Credits:1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of Basic mathematics, including calculus, complex numbers, and elementary linear algebra, is required for this course.
Course Objective: To provide students with a strong foundation in signal processing concepts, including signal representation, system analysis, time- and frequency-domain techniques, and their applications in modern engineering systems.
.
Course Outcome: After completion of the course the student will be able:
CO1. Introduce fundamental concepts of signal processing, including signal classification and system properties.
CO2. To analyze signals using time-domain and frequency-domain techniques such as convolution and Fourier transforms.
CO2. To apply sampling and basic digital signal processing concepts to real-world signals.
CO3. To design and implement basic signal processing solutions using computational tools for engineering applications.

List of Practical

Practical No.	Contents	Teaching Hours.
1	Install and configure the Matlab environment for Signal Processing. Write a program to generate and visualize basic continuous-time and discrete-time signals.	**2**
2	Write a program to perform time shifting, time scaling, and reversal of signals.	**2**
3	Write a program to generate unit impulse, unit step, ramp, exponential, and sinusoidal signals.	**2**
4	Write a program to compute the convolution of two discrete-time signals and verify system response.	**2**
5	Write a program to analyze the impulse response and stability of a discrete-time system.	**2**
6	Write a program to compute and plot the Fourier Transform of a signal and analyze its spectrum.	**2**
7	Write a program to implement Discrete Fourier Transform (DFT) and Fast Fourier Transform (FFT).	**2**

8	Write a program to demonstrate the effect of sampling and aliasing on a signal.	**2**
9	Write a program to design and analyze a basic FIR digital filter.	**2**
10	Write a program to apply digital signal processing techniques to a real-world signal (audio or biomedical).	**2**
11	Project	**10**

Books and References:

Text Books

3. **Signals and Systems** by Oppenheim & Willsky: The classic for understanding the core concepts of signals and systems before diving deep into digital.
4. **Understanding Digital Signal Processing** by Richard G. Lyons: Praised for its clarity and practical approach, great for beginners.

Reference Books:

3. **Understanding Digital Signal Processing** by Richard G. Lyons: Highly recommended for grasping DSP fundamentals intuitively.
4. **The Scientist & Engineer's Guide to Digital Signal Processing** by Steven W. Smith: Excellent for beginners, available online.

Online Resources:

9. https://github.com/openlists/DSPResources
10. https://www.mathworks.com/academia/courseware/teaching-signal-processing-and-communications-with-matlab-and-simulink.html

Syllabus
Semester VII

Course code: CAS41MEL405 **Course category:** Major Elective		**Course name:** Graphical User Interface
Credits: 3	**Teaching scheme:** L-3 P-0	**Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Foundational design knowledge, and basic technical skills.		
Course Objectives: Teaching students to design and build user-friendly, efficient, and visually appealing interfaces by covering principles of Human-Computer Interaction (HCI), event-driven programming, usability.		
Course Outcomes: After completion of the course the student will be able to:		
CO1: Understand the fundamentals of Graphical User Interface design, including components, event-driven programming, and GUI architecture.		
CO2: Design user-friendly interfaces by applying principles of usability, accessibility, and human-computer interaction (HCI).		
CO3: Develop GUI-based applications using appropriate frameworks and tools (such as Java Swing, JavaFX, .NET, or Python Tkinter).		
CO4: Implement event handling, layout management, and interactive controls to create responsive and dynamic user interfaces.		
CO5: Evaluate and test GUI applications for usability, performance, and user experience, and improve them based on feedback.		

Contents

Unit	Content	Teaching hours
1	**Introduction** to UI/UX and Human-Computer Interaction (HCI) **History of Screen Design:** Evolution of interfaces from command line to modern GUIs. **Principles of User Interface Design:** Foundational guidelines for good design	10

	(e.g., visibility, learnability, efficiency, error prevention). **Human Characteristics in Design:** Understanding human information processing subsystems (perception, memory, attention) and their impact on design.	
2	**User Research and Information Architecture:** **Design Thinking Process:** A user-centric approach to problem-solving, including empathy and ideation. **User Research Methods:** Techniques like user interviews, surveys, and field studies to understand user needs. **Personas and User Journey Maps:** Creating representations of target users and mapping their interaction flows. **Information Architecture (IA):** Organizing and structuring information effectively using methods such as card sorting and sitemaps. **Task and User Flows:** Defining the sequence of actions a user takes to complete a goal.	10
3	**Interaction Design:** **Interaction Principles:** Designing how users interact with the system across different devices (e.g., desktop, mobile, touch-based). **Interface Elements:** GUI components like menus, buttons, text boxes, and controls. **Feedback and Guidance:** Providing proper system feedback, assistance, and error handling mechanisms. **Visual Design Principles:** **Colour Theory and Typography:** The psychological and practical application of colour and fonts in visual communication.	10
4	**Layout and Composition:** Principles of arranging elements to create balanced and visually pleasing screens (e.g., grid systems, alignment, hierarchy). **Evaluation and Testing:** **Usability Testing:** Planning and conducting tests with real users to measure the	10

	usability of the interface. **Heuristic Evaluation:** Using established frameworks (like Nielsen's heuristics) to identify usability problems. **Accessibility:** Designing for inclusive access, often following guidelines like WCAG (Web Content Accessibility Guidelines).	
5	**Implementation:** **GUI Software Programming:** Hands-on experience implementing interfaces using programming languages (e.g., JavaScript, HTML/CSS, event handling, DOM manipulation). **Software Tools:** Proficiency in industry-standard design and prototyping software.	05

Textbooks:
1) Designing the User Interface: Strategies for Effective Human-Computer Interaction Ben Shneiderman and Catherine Plaisant (later editions include Maxine Cohen and Steven Jacobs), Pearson Education (Addison-Wesley).
2) Interaction Design: Beyond Human-Computer Interaction Yvonne Rogers, Helen Sharp, and Jennifer Preece, John Wiley & Sons.
Reference Books:
1) Designing the User Interface: Strategies for Effective Human-Computer Interaction by Ben Shneiderman, Catherine Plaisant, et al.
2) Human-Computer Interaction by Alan Dix, Janet Finlay, Gregory D. Abowd, and Russell Beale.

Syllabus
Semester-VIII

Course Code: CAS41MEL406 **Course Category:** Major Elective	**Course Name:** Biometrics Technology	
Credits: 3	**Teaching Scheme:**L-3 P-0	**Evaluation Scheme:** CA-60, ESE-40
Pre-requisites: Basic knowledge of Biometric authentication process, Awareness on different biometric devices		
Course Objectives: 1. This course introduces the fundamentals of biometric systems and multimodal recognition. 2. Understand the Biometric technology including the definition, terminologies used, parameters and basic features. 3. It also covers feature extraction, fusion techniques, and performance evaluation for reliable applications.		
Course Outcomes:		
CO1: Understand the fundamentals, evolution, components, and applications of biometric systems.		
CO2: Describe various biometric mechanisms.		
CO3: Understand Biometric recognition systems for human identification and recognition and purpose using Various biometric traits.		
CO4: categorizes biometric applications and technologies to real time Problems.		
CO5: Analyze multibiometric systems, their architectures, and sources of multiple biometric evidence.		

Course Contents –

Unit	Content	Teaching hours
1	**Basic of Biometric:** Introduction To Biometrics, Background And Evolution Of Biometric, What Is Biometric Recognition, Types Of Biometric-Physiological Biometrics: Face, Fingerprint, And Iris, Palm Etc. Behavioral Biometrics: Signature, Voice, Gait, Keystroke Etc. Biometric System Process-Enrollment, Verification, Identification. Component Of Biometric System, Biometric System Functionality, Applications Of Biometric Systems.	9
2	**Fingerprint Recognition :**Introduction, Friction Ridge Pattern, Features, Level 1 Features, Level 2 Features, Level 3 Features, Additional Features , Fingerprint Acquisition, Sensing Techniques, Image Quality, Feature Extraction- Ridge Extraction ,Minutiae Extraction.	9
3	**Face Recognition:** Introduction, Psychology Of Face Recognition, Facial Features, Level 1, Level 2, Level, Design of Face Recognition System, Image Acquisition Sensors, Face Detection and Feature Extraction Techniques.	9
4	**Iris Recognition:** Introduction, Anatomy of human eye, Design of an Iris Recognition System, Image Acquisition, Iris Segmentation, Normalization, Feature extraction, Performance Evaluation.	9
5	**Multibiometrics:** Introduction, Sources of Multiple Evidence, Multi-sensor systems, Multi-algorithm systems, Multi-instance systems, Multi-sample systems, Multimodal systems, Acquisition and Processing Architecture, Acquisition sequence, Processing sequence. Datasets. **Levels of Fusion**-Sensor-level fusion, Feature-level fusion, Score-level fusion, Fusion Techniques.	9

Text Books: 1. Anil K. Jain, Arun A. Ross, Karthik Nandakumar, Introduction to biometrics, Springer, 2011. 2. Anil K Jain, Patrick Flynn and Arun A Ross, Handbook of Biometrics, Springer, USA,2010.
Reference Books: 1. John R Vacca, Biometric Technologies and Verification Systems, Elsevier, USA, 2007. 2. Ross, Arun A., Anil K. Jain, and Karthik Nandakumar. *Handbook of multibiometrics*. Boston, MA: Springer US, 2006.
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus
Semester VII

Course code: CAS41MEP405 **Course name:** Practical Based on Graphical User Interface **Course category:** Major Elective
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: A blend of theoretical knowledge in design principles and practical skills in programming and using development tools.
Course Objectives: Applying design principles, developing functional interfaces using programming tools, and evaluating usability from a user perspective.
Course Outcomes: After completion of the course the student will be able to:
CO1: Understand the fundamentals of Graphical User Interface design, including components, event-driven programming, and GUI architecture.
CO2: Design user-friendly interfaces by applying principles of usability, accessibility, and human-computer interaction (HCI).
CO3: Develop GUI-based applications using appropriate frameworks and tools (such as Java Swing, JavaFX, .NET, or Python Tkinter).
CO4: Implement event handling, layout management, and interactive controls to create responsive and dynamic user interfaces.
CO5: Evaluate and test GUI applications for usability, performance, and user experience, and improve them based on feedback.

Contents

Practical No.	Content	Teaching hours
1	Create a basic window or form with a text box or label to display "Hello World".	2
2	Design and implement a simple calculator using buttons and text fields to perform addition, subtraction, multiplication, and division.	2
3	Develop an application that uses various controls such as checkboxes, radio buttons, and combo boxes, demonstrating how to retrieve and use their values.	2
4	Write programs that respond to user actions, such as button clicks, mouse movements, or keyboard input, to change the interface or display information.	2
5	Build an application that moves items between two list boxes or adds/removes items dynamically based on user input.	2
6	Create a simple login form and implement error handling and validation for user input (e.g., ensuring fields are not empty, checking password strength, using an error provider control).	2
7	Create a functional clock or stopwatch application using a timer component to update the display at regular intervals.	2

8	Develop a basic text editor with file handling capabilities (open, save, new file) and menu options.	2
9	Build a simple gallery application that allows browsing images in a folder using file dialogs.	2
10	Create applications that use Multiple Document Interface (MDI) forms, demonstrating how to open and manage child windows within a parent container.	2
11	Case Study.	2

Textbooks:
3) GUI Development in C#: Modern Interface Design | Create 12 Desktop Applications | WPF & XAML Projects by Emily Dawson
4) Practical UI by Adham Dannaway.
Handbooks:
3) The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques by Wilbert O. Galitz
4) Designing Interfaces: Patterns for Effective Interaction Design by Jenifer Tidwell.

Syllabus

Semester-VIII

Course Code: CAS41MEP406 **Course Name:** Practical based on Biometrics Technology **Course Category:** Major Elective
Credits: 1 **Teaching Scheme:**L-0,P-2 **Evaluation Scheme**: CA-30, ESE-20
Pre-requisites: Understanding of Digital Image Processing concepts, Basic programming skills in Python or MATLAB, Familiarity with matrices, vectors, and simple statistics
Course Objectives: The objective of this course is to introduce students to learn the fundamental concepts of biometric systems and human recognition using face, fingerprint, and iris traits. It also aims to develop an understanding of image processing, feature extraction, and multimodal fusion techniques used for secure and reliable identification systems.
Course Outcomes: At the end of the course, the students will be able to -
CO1: Understand the basic concepts, architecture, and working of biometric systems for human identification and recognition.
CO2: Analyze unimodal biometric techniques for face, fingerprint, and iris recognition.
CO3: Apply image processing techniques for biometric image preprocessing and feature extraction.
CO4: Design and implement multibiometric systems using different levels of fusion and fusion techniques.
CO5: Evaluate the performance of biometric and multibiometric systems using standard performance metrics.

List of Practicals:

Sr.No.	Title of the Experiment	Practical Hours
1	Study the basic concepts, components, and working of a biometric system.	2
2	Perform basic image operations Read and display a biometric image (face/fingerprint/iris).	2
3	Perform fingerprint image acquisition and analyze fingerprint image quality.	2
4	Extract fingerprint features such as ridge patterns and minutiae points.	2
5	Read and display face images from a dataset.	2
6	Detect facial region from an image.	2
7	Crop and align detected face images.	2
8	Read and display iris (eye) images from a dataset.	2
9	Locate the iris region in an eye image.	2
10	Segment the iris region by masking	2
11	Project	10

Text Books: 1. Anil K. Jain, Arun A. Ross, Karthik Nandakumar, Introduction to biometrics, Springer, 2011. 2. Anil K Jain, Patrick Flynn and Arun A Ross, Handbook of Biometrics, Springer, USA,2010.
Reference Books: 1. John R Vacca, Biometric Technologies and Verification Systems, Elsevier, USA, 2007. 2. Ross, Arun A., Anil K. Jain, and Karthik Nandakumar. *Handbook of multibiometrics*. Boston, MA: Springer US, 2006.
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus
Semester VII

Course code: CAS41RML401 **Course Name**: Research Methodology **Course Category**: Research Methodology
Credits:3 **Teaching scheme:** L-**3** P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic understanding of subject fundamentals, elementary statistics, and academic reading and writing skills.
Objective: To develop an understanding of research methodology principles and techniques applicable to problem identification, analysis.
Course Outcome: After completion of the course the student will be able:
CO1: To develops the ability to identify research problems and formulate objectives and hypotheses.
CO2: To familiarizes students with research design, data collection, and analysis techniques relevant to robotics.
CO3: To enhances skills in technical writing, documentation, and research ethics.
CO4: Analyze and apply statistical techniques such as data processing, descriptive statistics, correlation, regression, and hypothesis testing to interpret and validate research data.
CO5: Evaluate research findings and create a structured research report or mini proposal using appropriate referencing styles, ethical practices, and computer-based research tools.

Unit	Contents	Teaching Hours.
1	**Introduction to Research &Research Process :**Meaning of Research, objectives, motivation, and significance of research, Types of research (basic/applied, qualitative/quantitative, conceptual/empirical) Research methods vs. research methodology, Research Approaches, Significance of Research, Scientific method and research ethics, Steps in the research process, Criteria of Good Research, Problems Encountered by Researchers in India	09
2	**Research Problem Formulation & Research Design:** Identification and definition of research problems in engineering/robotics (Meaning and sources of research problems, Criteria for selecting a research problem, Defining and formulating research problems), Review of literature and research gap identification, Formulation of objectives and hypotheses, Research design: exploratory, descriptive, experimental, Features of a good research design	09
3	**Sampling Design & Data Collection Methods:** Sampling concepts, population, sample, sampling errors , Probability and non-probability sampling techniques (Sample size (basic concept)Types of sampling errors (Sampling error, Non-sampling error, Bias and sources of bias), Methods of data collection: observation, interview, questionnaire, experiments, Selection of appropriate sampling and data collection methods (Nature of research problem, Objectives of the study, Time, cost, and resource constraints, Accuracy and reliability requirements, Relevance to robotics and engineering research)	09
4	**Data Processing, Analysis & Hypothesis Testing:** Data processing (Editing, coding, classification, tabulation), Descriptive statistics) Measures of central tendency, Measures of dispersion, Correlation and regression, Hypothesis testing (Concepts and Procedure, parametric and non-parametric tests), Interpretation of Statistical results.	09

5	**Research Reporting:** Interpretation of research results, Research report writing (Structure of research report technical paper, and project report), Referencing styles and bibliography, Plagiarism and ethical issues in research, Role of computers in research (Data analysis tools, Documentation and presentation tools), Preparation of a mini research proposal / report	09

Text Books :1.C.R. Kothari — Research Methodology: Methods & Techniques 2.Ranjit Kumar — Research Methodology: A Step-by-Step Guide for Beginners
Reference Books: 1.Garg, B.L., Karadia, R., Agarwal, R., & Agarwal, U.K. — An Introduction to Research Methodology 2.Paul D. Leedy& Jeanne Ellis Ormrod — Practical Research: Planning and Design
Online Resources: 1. Alison – Essentials of Research Methodology

Syllabus
Semester-VII

Course Code: CAS41RMP401 **Course Name:** Practical Based on Research Methodology **Course Category:** Research Methodology
Credits: 1 **Teaching Scheme:**L-0, P-**2** **Evaluation Scheme**:CA-30,ESE-20
Pre-requisites: Research Methodology practical is a basic understanding of research design, data collection methods, and fundamental statistical and analytical tools.
Course Objectives: The Research Methodology course is to equip students with the skills to design, conduct, analyze, and interpret research systematically and ethically.
Course Outcomes: At the end of the course, the students will be able to -
CO1: To develops the ability to identify research problems and formulate objectives and hypotheses.
CO2: To familiarizes students with research design, data collection, and analysis techniques relevant to robotics.
CO3: To enhances skills in technical writing, documentation, and research ethics.
CO4: Analyze and apply statistical techniques such as data processing, descriptive statistics, correlation, regression, and hypothesis testing to interpret and validate research data.
CO5: Evaluate research findings and create a structured research report or mini proposal using appropriate referencing styles, ethical practices, and computer-based research tools.

List of Practicals:

Sr.No.	Title of the Experiment
1	Identify research papers from journals/conferences and prepare a structured literature review.
2	Select a domain in Computer fraternity and define a clear research problem with objectives.
3	Write research objectives and formulate null and alternative hypotheses.
4	Develop a suitable research design for a selected problem (exploratory/descriptive/experimental).
5	Identify population, sample size, sampling technique, and justify the choice.
6	Design a questionnaire for data collection related to a Computer fraternity problem.
7	Collect sample data using questionnaire/observation and enter data in spreadsheet/software.
8	Perform basic statistical analysis (mean, median, standard deviation, graphs).
9	Apply an appropriate statistical test (t-test / chi-square) and interpret results.
10	Prepare and present a mini research report standard research paper format.

Text Book: 1**. Practical Research: Planning and Design –** Paul D. Leedy & Jeanne Ellis Ormrod
Reference Book: 1. Research Methodology: Methods and Techniques – C.R. Kothari

B.C.A.(Hons. With Research)
Semester: EIGHT

Syllabus
Semester-VIII

Course Code:CAS41MML408 **Course Name:** Human Computer Interface **Course Category:** Major Mandatory
Credits: 3 **Teaching Scheme:**L-3 P-0 **Evaluation Scheme:** CA-60, ESE-40
Pre-requisites: Basic programming knowledge, basic computer skills, and understanding of software applications.
Course Objectives: To understand human–computer interaction concepts, design usable interfaces, and explore evaluation techniques and emerging HCI technologies.
Course Outcomes: At the end of the course, the students will be able to -
CO1: Understand human capabilities, interaction styles, and interface paradigms.
CO2: Apply interaction design principles and usability engineering in software processes
CO3: Use evaluation techniques and cognitive models to assess usability
CO4: Analyze socio-organizational issues, task models, and design dialogues
CO5: Explore advanced interaction models and emerging technologies in HCI.

Course Contents –

Unit	Content	Teaching hours
1	**Foundations of Human–Computer Interaction**: The Human: capabilities, limitations, cognitive aspects, The Computer: input/output devices, interaction devices, The Interaction: interaction styles, interaction loop, Interaction Paradigms: Command-based interaction, Direct manipulation, Menu-based systems, Natural language interaction	8
2	I**nteraction Design and Software Process:** Interaction Design Basics, User-Centred Design Principles, HCI in Software Development Process, Design Lifecycle & Usability Engineering, Design Rules: Principles, Standards, Guidelines, Golden rules of interface design. Implementation Support: Toolkits, User Interface Management Systems (UIMS)	10
3	**Evaluation and User-Centred Design:** Evaluation Techniques: Usability Testing, Heuristic Evaluation, Cognitive Walkthrough, Universal Design & Accessibility, User Support: Help Systems, Documentation & Online Help, Cognitive Models: GOMS Model, Keystroke-Level Model (KLM)	10

4	**Organisational and Task-Oriented Design:** Socio-Organisational Issues, Stakeholder Requirements Analysis, Communication and Collaboration Models, Computer-Supported Cooperative Work (CSCW), Task Analysis: Hierarchical Task Analysis, Use Cases & Scenarios. Dialogue Notations & Design: State Transition Diagrams, Flowcharts, Dialogue Design Techniques	8
5	**Advanced Interaction and Emerging Technologies:** Models of the System: Conceptual Models, System Models. Modelling Rich Interaction, Groupware & Collaborative Systems, Ubiquitous Computing & Augmented Reality, Hypertext, Multimedia & the World Wide Web, Applications and Future Trends in HCI	9

Text Books: 1.Alan Dix, Janet Finlay, Gregory Abowd & Russell Beale – Human–Computer Interaction, Pearson 2.Jenny Preece, Yvonne Rogers, Helen Sharp – Interaction Design: Beyond Human–Computer Interaction, Wiley
Reference Books: 1.Donald Norman – *The Design of Everyday Things*, Basic Books 2.Ben Shneiderman & Catherine Plaisant – *Designing the User Interface: Strategies for Effective Human–Computer Interaction*, Pearson. 3.Amrita Vishwa Vidyapeetham HCI Notes / Syllabus (Online)
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus
Semester-VIII

Course Code: CAS41MML409 **Course Name:** Biomedical Image Processing **Course Category:** Major Mandatory
Credits: 3 **Teaching Scheme:**L-3 P-0 **Evaluation Scheme:** CA-60, ESE-40
Pre-requisites: Basic programming skills, basic mathematics, and fundamental computer knowledge.
Course Objectives: To understand medical images, apply image processing techniques, and explore healthcare applications and future trends.
Course Outcomes: At the end of the course, the students will be able to -
CO1: Understand modern medical imaging modalities, their applications, and recent technological advancements.
CO2: Apply enhancement techniques to current biomedical images and understand their limitations
CO3: Perform segmentation and extract meaningful features using current and AI-based methods.
CO4: Implement registration and classification techniques with AI and evaluate their performance.
CO5:Analyze real-world biomedical applications, explore emerging technologies, and discuss ethical considerations

Course Contents –

Unit	Content	Teaching hours
1	**Medical Imaging Modalities & Basics:** What are biomedical images Importance and applications in healthcare, Overview of medical imaging systems, Imaging modalities: X-ray, CT, MRI, Ultrasound, PET/SPECT, Properties of medical images; DICOM data format, Fundamentals of image acquisition, Recent hardware advances: high-resolution CT, portable ultrasound, AI-assisted imaging devices	9
2	**Fundamentals of Digital Image Processing & Enhancement:Digital** image representation: pixels, voxels, grid, sampling, Quantization and color models (RGB, grayscale),Basic image operations: brightness, contrast, Visual perception and quality measures, Point operations and histogram processing, Spatial domain enhancement: smoothing & sharpening filters, Frequency domain enhancement: low-pass/high-pass filtering, homomorphic filtering, Noise removal and adaptive/wavelet-based denoising,2D Discrete Fourier Transform (DFT) and basics of image transform	9
3	**Segmentation & Morphological Processing:** Thresholding (global & adaptive),Edge detection: Sobel, Canny, Prewitt, Region-based segmentation: region growing, splitting, merging, Morphological operations: dilation, erosion, opening, closing, boundary extraction, Feature extraction: shape & texture descriptors, Introduction to deep learning-based segmentation (U-Net, CNN)	9

4	**Medical Image Registration & Classification**: Geometric transformations Image registration fundamentals: rigid & non-rigid registration, Similarity measures for registration, Classification basics: supervised & unsupervised learning, Machine learning & deep learning classifiers, Evaluation metrics for segmentation and classification	9
5	**Applications & Future Trends:** Case studies: tumour detection, mammography/ultrasound enhancement, Multimodal image fusion, 3D visualization and volumetric imaging, AI/ML applications in medical imaging, AR/VR applications in surgery and diagnosis, Ethical AI, data privacy, and clinical relevance, Future research directions in biomedical imaging	9

Text Books: 1.Rafael C. Gonzalez & Richard E. Woods *Digital Image Processing*, 4th Edition, Pearso 2.Atam P. Dhawan *Medical Image Analysis*, Wiley-IEEE Press
Reference Books: 1.Paul Suetens *Fundamentals of Medical Imaging*, 3rd Edition, Cambridge University Press 2.Jerry L. Prince & Jonathan M. Links *Medical Imaging Signals and Systems*, Pearson 3.Anil K. Jain *Fundamentals of Digital Image Processing*, Prentice Hall 4.Richard Szeliski *Computer Vision: Algorithms and Applications*, Springer
Online Resources: 1.NPTEL / SWAYAM lectures.

Syllabus

Semester-VIII

Course Code: CAS41MMP408 **Course Name:** Practical Based on Human Computer Interface **Course Category:** Major Mandatory
Credits: 1 **Teaching Scheme:**L-0,P-2 **Evaluation Scheme**: CA-30, ESE-20
Pre-requisites: Basic programming, computer skills, and software fundamentals
Course Objectives: Learn HCI concepts, design usable interfaces, and explore usability and emerging technologies.
Course Outcomes: At the end of the course, the students will be able to -
CO1: Understand human-computer interaction and devices
CO2: Apply interaction design and usability principles
CO3: Evaluate interfaces using cognitive models and testing
CO4: Analyze tasks, dialogues, and organizational issues
CO5: Explore advanced HCI technologies, multimedia, and web interfaces

List of Practicals:

Sr.No.	Title of the Experiment	Practical Hours
1	Create a simple window with labels, buttons, and text fields.	2
2	Implement buttons that perform actions like changing text or displaying messages	2
3	Take input from the user (name, age, etc.) and display output (e.g., simple calculator).	2
4	Create a simple menu with options like File, Edit, and Help	2
5	Add tooltips or help messages for GUI elements.	2
6	Validate user inputs (numeric fields, email format, mandatory fields).	2
7	Load and display an image or a small audio/video file in the GUI	2
8	Create multiple GUI screens/windows and navigate between them using buttons.	2
9	Combine GUI elements to create a small interactive program (e.g., Quiz, To-Do List).	2
10	Design a GUI interface applying usability principles (layout, colors, feedback).	2
11	Project	10

Reference Book / Hand Books/ Lab Manual
1.Ben Shneiderman & Catherine Plaisant – *Designing the User Interface: Strategies for Effective HCI*, Pearson
2. Donald Norman – *The Design of Everyday Things*, Basic Books
3. Alan Dix, Janet Finlay, Gregory Abowd & Russell Beale – *Human–Computer Interaction*, Pearson

Syllabus
Semester-VIII

Course Code: CAS41MMP409 **Course Name:** Practical Based on Biomedical Image Processing **Course Category:** Major Mandatory
Credits: 1 **Teaching Scheme:**L-0, P-2 **Evaluation Scheme**: CA-30, ESE-20
Pre-requisites: Basic knowledge of programming (MATLAB / Python) and basic computer operations.
Course Objectives: To implement image processing techniques on medical images using MATLAB / Python.
Course Outcomes: At the end of the course, the students will be able to -
CO1: Understand biomedical image types and read/display medical images using MATLAB/Python
CO2: Apply digital image representation concepts and basic image enhancement techniques
CO3: Perform histogram-based processing and noise removal using filtering methods.
CO4: Apply frequency domain processing, edge detection, segmentation, and morphological operations.
CO5: Analyze medical images and implement a simple biomedical image processing application/project.

List of Practicals:

Sr.No.	Title of the Experiment	Practical Hours
1	To read and display biomedical images (X-ray/CT/MRI).	2
2	To study digital image representation and grayscale conversion.	2
3	To perform brightness and contrast enhancement on medical images.	2
4	To plot image histogram and apply histogram equalization.	2
5	To add noise and remove it using filtering techniques	2
6	To apply spatial domain filters for smoothing and sharpening.	2
7	To apply frequency domain processing using 2D FFT.	2
8	To detect edges using Sobel, Prewitt, and Canny operators.	2
9	To perform image segmentation using thresholding techniques	2
10	To apply morphological operations on segmented images.	2
11	Project(Ex-Tumor or abnormal region detection from medical images using Matlab/Python)	10

Reference Book / Hand Books/ Lab Manual
1.**Rafael C. Gonzalez and Richard E. Woods** – *Digital Image Processing*, Pearson
2. **Anil K. Jain** – *Fundamentals of Digital Image Processing*, Pearson
3. MATLAB / Python Image Processing Documentation and Lab Manuals

Syllabus
Semester VIII

Course code: CAS41MEL407 **Course category:** Major Elective		**Course Name:** Artificial Intelligence
Credits: 3	**Teaching scheme:** L-3 P-0	**Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basic knowledge of algorithms & data structures.		
Course Objectives: Understand basic principles, techniques, and applications of Artificial Intelligence and Problem solving, inference, perception, knowledge representation, and learning.		
Course Outcomes: At the end of the course, the students will be able to -		
CO 1: Understanding of the history of artificial intelligence (AI) and its foundations.		
CO 2: Apply basic principles of AI in solutions that require problem solving, inference		
CO 3: Perception, knowledge representation, and learning.		
CO 4: Demonstrate awareness and a fundamental understanding of various applications of AI		
CO 5: Develop Knowledge based System		

Contents –

Unit	Content	Teaching hours
1	**Introduction:-**Introduction to Artificial Intelligence, Historical Background, What is Intelligence, Depth First Search (DFS), Breadth First Search (BFS), Comparison of Depth First Search and Breadth First Search, Quality of Solution	10
2	**Heuristic Search:-** Heuristic Functions, Best First Search, Hill Climbing, Local Maxima, Solution Space Search, Variable Neighbourhood Decent, Beam Search, Tabu Search, Peak to Peak Method.	10
3	**Randomized Search and Emergent Systems**:-Iterated Hill Climbing, Simulated Annealing, Genetic Algorithm, The Travelling Salesman Problem, Neural Network, Emergent System, Ant Colony Optimization	10
4	**Finding Optimal Paths:-**Brute Force, Branch and Bound, Refinement Search, Dijkstra's Algorithm, Algorithm of A*, Admissibility of A*, Recursive Best First Search (RBFS), Pruning the CLOSED List, Pruning the OPEN List, Divide and Conquer Beam Stack Search	10
5	**Knowledge Representation:-**Knowledge Representation, Representation and Mapping, Approaches to Knowledge Representation, Issues in Knowledge Representation, Predicate Logic, Representing Knowledge Using Rules	5

Text Books:
1. A First Course in Artificial Intelligence Deepak Khemani McGraw-Hill Education.
2. Artificial Intelligence A Modern Approach by stuart Russell, peter Norvig, third edition.

Reference Books: 1.Artificial Intelligence Elaine Rich Tata McGraw-Hill Education
2. Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter NorvigPublisher: Pearson ISBN-13: 978-0134610993

Semester VIII

Course code: CAS41MEL408 **Course Name**: Big Data Analytics **Course Category**: Major Elective
Credits: 3 **Teaching scheme:** L-3 P-0 **Evaluation scheme:** CA–60, ESE–40
Pre-requisites: Basics Data and Data Analytics.
Course Objective:
To enable students to apply various analytical techniques and tools for processing and deriving insights from large datasets.
Course Outcome: After completion of the course the student will be able:
CO1. Analyze the characteristics and challenges of Big Data and its impact on various domains.
CO2. Implement solutions for Big Data storage and processing using frameworks such as Hadoop and Spark.
CO3. Apply appropriate Big Data analytic techniques to extract meaningful insights from large datasets.
CO4: Utilize appropriate Big Data analytic techniques and tools to extract meaningful insights from large datasets.
CO5: Evaluate Big Data case studies, propose solutions for complex analytical problems, and consider ethical implications.

Contents:

Unit	Contents	Teaching Hours.
I	**Introduction to Big Data and its Ecosystem:** - Understanding Big Data: Characteristics, challenges, and opportunities. - Big Data Architectures: Lambda and Kappa architectures. - Introduction to Hadoop: HDFS architecture, MapReduce programming model. - Overview of the Hadoop Ecosystem: YARN, Hive, Pig, Sqoop.	8
II	**NoSQL Databases** - Introduction to NoSQL: Motivations, CAP theorem, types of NoSQL databases. - Document Databases: MongoDB (data model, basic operations). - Column-Family Databases: Cassandra (data model, basic operations) - Comparison of NoSQL databases with traditional RDBMS.	9
III	**Data Ingestion and Warehousing** : - Data Ingestion: Batch vs. Stream processing, data connectors, real-time data ingestion. - Data Warehousing concepts in a Big Data context: Data Lake vs. Data Warehouse. - Data Governance and Metadata Management in Big Data environments. - ETL vs. ELT in Big Data.	9

IV	**Big Data Processing with Spark** : - Introduction to Apache Spark: RDDs, Spark Architecture, comparison with MapReduce. - Spark Core: Transformations and Actions, lazy evaluation. - Spark SQL: Working with DataFrames and Datasets, Catalyst Optimizer. - Spark Streaming/Structured Streaming: Real-time data processing concepts. - Introduction to PySpark/Spark Scala.	10
V	**Big Data Analytics Techniques and Applications** - Data Preprocessing and Cleaning for Big Data. - Machine Learning with Big Data: Overview of MLlib, common algorithms (clustering, classification). - Big Data Visualization: Tools and techniques for presenting insights - Ethical considerations and privacy in Big Data.	9

Books and References:

Text Books **1.** Kumar, J., Kumar, A., & Kumar, R.. *Big Data and Analytics: The key concepts and practical applications of big data analytics*. BPB Publications **2.** Demchenko, Y., Cuadrado-Gallego, J. J., Chertov, O., & Nordström, L.. *Big Data Infrastructure Technologies for Data Analytics*
Reference Books: **1.** White, T.. *Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale* (4th ed.). O'Reilly Media **2.** Rajaraman, A., & Ullman, J. D.. *Mining of Massive Datasets*. Cambridge University Press
Online Resources: **1.** Official documentation for Apache Hadoop, Apache Spark, MongoDB, Cassandra, etc.

Syllabus
Semester VIII

Course code: CAS41MEP407 **Course name:** Practical Based on Artificial Intelligence **Course category:** Major Elective		
Credits: 1	**Teaching scheme:** L-0 P-2	**Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basic knowledge of algorithms & data structures.		
Course Objectives: Understand basic principles, techniques, and applications of Artificial Intelligence and Problem solving, inference, perception, knowledge representation, and learning.		
Course Outcomes: At the end of the course, the students will be able to -		
CO1: Understanding of the history of artificial intelligence (AI) and its foundations.		
CO2: Apply basic principles of AI in solutions that require problem solving, inference		
CO3: Perception, knowledge representation, and learning.		
CO4: Demonstrate awareness and a fundamental understanding of various applications of AI		
CO5: Develop Knowledge based System		

List of Practical:

Sr.No.	Title of the Experiment	Practical hours
1	Install and configure the Python Program to implement the Breadth First Search algorithm in Python	2
2	Program to implement the Depth First Search algorithm in Python	2
3	Design a program to solve water jug problem and implement it in python	2
4	Design a program to solve the Tower of Hanoi problem using Python	2
5	Design a program to solve Best First Search Algorithm using Heuristic Search techniques	2
6	Design a program to implement Randomized Search	2
7	Program to implement solution of The Travelling Salesman Problem	2
8	Program to implement Ant Colony Optimization Technique	2
9	Design solution for shortest path for finding two geographical locations	2
10	Design solution for Knowledge Representation using Predicate Logic	2
11	Design Project on AI Applications	10

Reference Book / Hand Books/ Lab Manual

1.A First Course in Artificial Intelligence Deepak Khemani McGraw-Hill

Education
2.Artificial Intelligence Elaine Rich Tata McGraw-Hill Education Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter NorvigPublisher: Pearson ISBN-13: 978-0134610993

Syllabus
Semester-VIII

Course code: CAS41MEP404 **Course Name**: Practical Based on Big Data Analytics **Course Category**: Major Elective
Credits: 1 **Teaching scheme:** L-0 P-2 **Evaluation scheme:** CA–30, ESE–20
Pre-requisites: Basics Data and Data Analytics.
Course Objective:
The main objective is to solve the customer's requirements.
Course Outcome: After completion of the course the student will be able:
CO1: Set up and interact with a Big Data environment, including Hadoop HDFS and basic MapReduce jobs.
CO2: Implement data storage and retrieval operations using NoSQL databases like MongoDB and Cassandra.
CO3: Write and execute data processing scripts using Apache Spark for various data manipulation tasks.
CO4: Perform basic data ingestion and preprocessing steps on large datasets.
CO5: Apply fundamental Big Data analytical techniques and visualize results using appropriate tools.

Contents:

Practical No.	Description	Teaching Hours
1	Set up a single-node Hadoop cluster, Perform basic HDFS commands: put, get, ls, mkdir, cat.	2
2	Write a basic MapReduce program (e.g., Word Count) in Java or Python	2
3	Compile and execute the MapReduce job on the Hadoop cluster., Analyze the output files generated by the MapReduce job	2
4	Install and configure MongoDB. Create a database and collections.	2
5	Insert, find, update, and delete documents using MongoDB shell commands or a Python client.	2
6	Set up a single-node Cassandra instance. Create keyspaces and tables.	2
7	Insert data, query data using CQL. Understand the basic differences between MongoDB and Cassandra in practical.	2
8	Set up Spark in local mode. Load data (e.g., from a text file) into an RDD.	2

9	Perform basic RDD transformations (map, filter, flatMap) and actions (count, collect).	2
10	Implement a simple data cleaning task using RDDs.	2
11	Mini Project / Case Study	10

Books and References:

Text Books

1. Kumar, J., Kumar, A., & Kumar, R.. *Big Data and Analytics: The key concepts and practical applications of big data analytics*. BPB Publications

2. Demchenko, Y., Cuadrado-Gallego, J. J., Chertov, O., & Nordström, L.. *Big Data Infrastructure Technologies for Data Analytics*

Reference Books:

1. White, T.. *Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale* (4th ed.). O'Reilly Media

2. Rajaraman, A., & Ullman, J. D.. *Mining of Massive Datasets*. Cambridge University Press

Online Resources:

1. Official documentation for Apache Hadoop, Apache Spark, MongoDB, Cassandra, etc.