KANNUR UNIVERSITY

(Abstract)

M Sc. Computer Science Programme in the Department of Information Technology, Mangattuparamba Campus - Revised Scheme and Syllabus(I st Semester only)- Approved-Implemented w.e.f. 2023 admission--Orders issued

ACADEMIC C SECTION

ACAD C/ACAD C1/23405/2023 Dated: 23.11.2023

Read:-1. U. O. No. ACAD C/ACAD C3/22373/2019 dtd.12.09.2023

2. Circular No. dated ACAD C/ACAD C3/22373/2019 dated 12/09/2023

3. Email dated 09.11.2023 from the Head, Department of Information Technology, Mangattuparamba Campus.

4.Minutes of the meeting of the Department Council held on 04.10.2023

ORDER

1.The revised Regulations for Post Graduate Programmes under Choice Based Credit and Semester System in the University Teaching Departments/Schools were implemented w.e.f. 2023 admissions vide paper read(1) above.

2. As per paper read (2) above, Heads of all Teaching Departments were requested to submit the revised Syllabus in accordance with the approved regulations along with a copy of the Department Council Minutes

3. As per paper read (3) above, the Head, Department of Information Technology, Mangattuparamba Campus submitted the Scheme & Syllabus (Ist Semester only) of M Sc. Computer Science Programme to be implemented in the University Teaching Department w. e. f. 2023 admissions.

4. Department Council vide the paper read (4) above approved the aforementioned Scheme & Syllabus of M Sc. Computer Science Programme to be implemented in the Dept. of Information Technology of the University w. e. f. 2023 admission.

5.The Vice Chancellor after considering the matter in detail, and in exercise of the powers of the Academic Council conferred under section 11(1), Chapter III of Kannur University Act 1996, approved the **Scheme & Syllabus (I st Semester only) of M Sc. Computer Science Programme and accorded sanction to implement the same in the Department of Information Technology, Mangattuparamba Campus of the University w.e.f. 2023 admissions, subject to report to the Academic Council.**

6.The Scheme & Syllabus (I st Semester Only) of M.Sc. Computer Science Programme, under Choice Based Credit Semester System implemented in the Department of Information Technology, Mangattuparamba Campus w. e. f. 2023 admission, is appended and uploaded in the University Web Site.(www.kannuruniversity.ac.in)

7.Orders are issued accordingly.

Sd/-

Narayanadas K
DEPUTY REGISTRAR (ACAD)
For REGISTRAR

To: 1.The Head, Department of Information Technology, Mangattuparamba Campus.
2. Convener, Curriculum Committee.

Copy To: 1. The Examination branch (through PA to CE)

2. PS to VC/ PA to PVC/PA to R

3. DR/AR1/AR II (Acad), EXCI, EP IV

4. Web Manager (for uploading in the website)

5. Computer Programmer

6.SF/DF/FC

Forwarded / By Order

SECTION OFFICER

KANNUR UNIVERSITY
CIVIL STATION P.O.
KANNUR
Pin-670 002
THAVAKKARA

KANNUR UNIVERSITY

(Abstract)

M. Sc. Computer Science Programme in the Department of Information Technology, Mangattuparamba Campus - II nd, III rd and IV th Semester syllabus - Approved- Implemented w.e.f. 2023 admission--Orders issued

ACADEMIC C SECTION

ACAD C/ACAD C1/23405/2023 Dated: 05.04.2024

Read:-1. U. O. No. ACAD C/ACAD C1/23405/2023 dated 23.11.2023

2. Circular No. dated ACAD C/ACAD C3/22373/2019 dated 01/02/2024

3. Email dated 28.02.2024 from the Head, Department of Information Technology, Mangattuparamba Campus.

4.Minutes of the meeting of the Department Council held on 13.02.2024

ORDER

1.The Scheme & Syllabus(Ist Semester only) of M Sc Computer Science Programme under Choice Based Credit and Semester System in the Department of Information Technology were implemented w.e.f. 2023 admissions vide paper read(1) above.

2. As per paper read (2) above, Heads of Teaching Departments who have not yet submitted the completed syllabus of PG Programmes were requested to submit the Syllabus of remaining semesters, prepared in tune with the approved Regulations / Scheme / Credit distribution table along with a copy of the Department Council Minutes

3. As per paper read (3) above, the Head, Department of Information Technology, Mangattuparamba Campus submitted the IInd, IIIrd& IVth semester Syllabus of M Sc. Computer Science Programme to be implemented in the University Teaching Department w. e. f. 2023 admissions.

4. Department Council vide the paper read (4) above approved the aforementioned Syllabus of M Sc. Computer Science Programme to be implemented in the Dept. of Information Technology of the University w. e. f. 2023 admission.

5.The Vice Chancellor after considering the matter in detail, and in exercise of the powers of the Academic Council conferred under section 11(1), Chapter III of Kannur University Act 1996, ***approved the IInd, IIIrd& IVth semester Syllabus of M Sc. Computer Science Programme and accorded sanction to implement the same in the Department of Information Technology, Mangattuparamba Campus of the University w.e.f. 2023 admissions, subject to report to the Academic Council.***

6.The Scheme & Syllabus of M.Sc. Computer Science Programme, under Choice Based Credit Semester System implemented in the Department of Information Technology, Mangattuparamba Campus w. e. f. 2023 admission, is appended and uploaded in the University Web Site.(www.kannuruniversity.ac.in)

7.Orders are issued accordingly.

Sd/-

Narayanadas K
DEPUTY REGISTRAR (ACAD)
For REGISTRAR

To: 1. The Head, Department of Information Technology, Mangattuparamba Campus.

2. Convener, Curriculum Committee.

Copy To: 1. The Examination branch (through PA to CE)

2. PS to VC/ PA to CE / PA to R

3. DR/AR1/AR II (Acad)

4. Web Manager (for uploading in the website)

5. Computer Programmer

6.EP IV/ EXC I Section

7 .SF/DF/FC

Forwarded / By Order

SECTION OFFICER

KANNUR UNIVERSITY
CIVIL STATION P.O.
KANNUR
Pin-670 002
THAVAKKARA

DEPARTMENT OF INFORMATION TECHNOLOGY

KANNUR UNIVERSITY

DEGREE OF

MSc. COMPUTER SCIENCE

(CHOICE BASED CREDIT AND SEMESTER SYSTEM)

SYLLABUS

(FOR THE STUDENTS ADMITTED FROM THE ACADEMIC YEAR 2023 – 24 ONWARDS)

REGULATIONS FOR THE DEGREE OF

MSc. COMPUTER SCIENCE

FOR THE STUDENTS ADMITTED FROM THE ACADEMIC YEAR 2023 – 24 ONWARDS

PROGRAMME SPECIFIC OUTCOMES

SL #	Outcome
PSO1	Familiar with the entrenched concepts of Computer Science and Applications.
PSO2	Attain skills to design Algorithms and Programs.
PSO3	Design, build up, put into practice and test software systems to meet the given specifications.
PSO4	Enhance the knowledge about emerging topics in Computer Science
PSO5	Attain prerequisite skills to address research problems in computer science.
PSO6	Furnish the candidate to do the real time jobs linked with Information Technology and Computer Application.

1. ADMISSION

Admission to the MSc Computer Science programme will be carried out as per the Kannur University PG regulations – 2023.

Eligibility for Admission:

Basic qualifications: A pass in BCA/ Bachelor Degree in Computer Science/ Mathematics/ Physics/ Statistics/Chemistry/ Engineering or equivalent degree.

2. PROGRAMME STRUCTURE

Duration of the MSc Computer Science programme shall be 2 years, divided into 4 semesters. Each semester shall have 18 weeks. The minimum duration for the completion of the MSc Computer Science programme is four (4) semesters from the date of registration. The maximum period of completion is eight semesters (4 years) from the date of registration. Every student have to opt the various categories of course [Discipline Specific Core (DSC), Discipline Specific Electives (DSE), Value Added Courses (VAC), Skill Enhancement Courses (SEC), Ability Enhancement Courses (AEC), Multi-Disciplinary Courses (MDC), MOOC and Inter Disciplinary Courses (IDC)] as per Kannur University PG Regulations-2023.

For MSc Computer Science programme, all students have to take a MOOC (of minimum 2 credits) and submit the course completion certificate obtained from the MOOC provider at the end of third semester. Credit earned from the MOOC will be over and above the minimum credit required for completion of MSc Computer Science Degree (Refer Kannur University PG Regulations-2023).

For the MSc Computer Science programme, a minor project work with 2 credits (offered in the third semester) is opted by the department.

The minimum credits required for the successful completion of the MSc Computer Science programme will be 84. Every student should earn a minimum of 8 credits offered by the other departments.

For all core courses an additional module (called as Module X) is added. Portions of this module need not be discussed by the faculty during the designated classroom hours for the given course. However components of continuous evaluation such as assignment / seminar / viva may be based on the contents of Module X. Module X has to be considered as additional reading and self-exploration by the students. For end semester examinations, contents of Module X will not be considered.

MOOC

Kannur University PG Regulations-2023 will be applicable for MOOC.

For MSc Computer Science programme, all students have to take a MOOC (of minimum 2 credits) and submit the course completion certificate obtained from the MOOC provider at the end of third semester. Department council will prepare a list of MOOCs from the online courses offered by the NPTEL / any other MOOC provider approved by the council from time to time at the beginning of semesters 1, 2 and 3. Students have to choose a MOOC from this list and earn his/her certificate of completion as per the guidelines stipulated by the MOOC provider concerned from time to time.The credit earned by the student for the MOOC will be counted as such without any normalization.

3. EVALUATION

Evaluation (Both CE and ESE) will be conducted as per the Kannur University PG Regulations- 2023 for all theory and practical courses. However the specific guidelines for conducting the evaluation of theory courses, practical courses, Mini project and Major project are mentioned below.

Continuous Evaluation For Theory Courses (DSC, DSE, VAC, IDC, MDC, AEC, SEC)

CE includes assignments, seminars, viva and periodic written examinations. This should be done by the faculty who manages the course.

The weightage of each component under CE for theory courses shall be in the following proportions:

Components	**% of Weightage**
Test papers (minimum two Continuous Evaluation Tests)	40
Assignments	20
Seminar / Viva	40

Table 1: Weightage of each Component under Continuous Evaluation for Theory Courses

Test Papers: There shall be a minimum of two test papers to be conducted for each course. If more than two test papers are conducted, then two best grades shall be taken for the award of CE grades. The dates of test papers shall be announced well in advance and the result should be displayed in the notice board. Tests such as multiple choice objective type and open text book test (online or offline mode) also can be opted for conducting the test papers.

Assignments: For each course at least one assignment (including practical assignments, if necessary) shall be assigned to the students. The mode of submission and assessment of the assignments shall be decided by the faculty concerned. Assignment works can be conducted either offline/ online mode (as per the decision taken by the faculty concerned).

Viva: Faculty concerned can assign topics for comprehension (based on any portions in the syllabus) and ask the students to appear for individual viva sessions as per a declared schedule. Viva sessions can be conducted either in online or offline mode (as per the decision taken by the faculty concerned).

Seminar: Faculty concerned can assign topics for comprehension (based on the course concerned) and ask the students to prepare seminars based on the topics assigned to them. Each student has to prepare the seminar content and present it. Mode of preparation, submission, and presentation can be specified by the faculty member concerned. Seminar sessions and contents can be submitted and conducted either in online or offline mode (as per the decision taken by the faculty concerned).

Technology Specific Electives (Elective V)

In the fourth semester an Elective course –Technology Specific Elective (Elective V) are meant to foster the students with tools and technologies that they need to know and make use in the design and development of software applications. The department council will prepare the list of elective courses to be offered for Elective V at the end of every third semester. Seminar/Report/ Case study implementation report of the specific technology mentioned in the elective should be submitted by each student for the evaluation. The mode of evaluation of this course shall be based on the presentation, report and viva. Both CE and ESE for this course will be conducted by the Department.

Continuous Evaluation For Practical Courses

The components of CE for practical courses are as follows:

Components	% of Weightage
Lab test (minimum 1)	40
Completion of the list of Lab assignments prescribed by the faculty	20
Periodical assessment of assignment in the Lab	40

Table 2: Weightage of each Component under Continuous Evaluation for Practical Courses

Evaluation For Minor Project Work

The components of CE and ESE for minor project work are as follows:

Components	% of Weightage
Understanding of the problem / Concepts	20
Adhering to methodology	15
Quality of presentation and demonstration	15
Quantum of work / effort	25
Organization and content of Project report	5
Viva based on Project	20

Table 3: Weightage of each Component for CE/ ESE for Minor Project Work

CE and ESE of the minor project work shall be done by a departmental committee constituted by the HOD. The committee should consist of a minimum of two faculty members, including the guide. Phases of evaluation and evaluation criteria for each phase shall be framed by the departmental committee.

Project Work

Project Work offered in the fourth semester currently has 8 credits. Project work has to be undertaken by all students. The project can be software development following all or some of the software development lifecycle or an R & D project. The hours allotted for project work may be clustered into a single slot so that students can do their work at a centre or location for a continuous period of time. The project work should be carried out in the department / Institution / industry / R & D organization of national repute. Project work shall be carried out under the supervision of a faculty member. If the student wishes to undertake his / her project outside the campus, then a co-guide shall be selected from the organization concerned. If the project work is of interdisciplinary nature, a co-guide shall be taken from the other department concerned. Every student should do the project individually and no grouping is allowed. The candidates are required to get the approval of the project synopsis from the supervisor in the department before the commencement of the project. A co-guide should be an expert in the area in which the student has chosen the project. At the end of the semester the candidate shall submit the project report (two bound copies and one soft copy) duly approved by the guide and co-guide for end semester evaluation. The project report shall be prepared according to the guidelines appended along with these regulations / guidelines. Students have to submit the copies of the reports that are approved by the project supervisor(s) before the last date fixed by the department.

The end semester evaluation of the project work shall be done by a board of at least two examiners, in which one should be an external expert. For the evaluation of the project work, the candidate must present the work before the board of examiners which will be followed by a Viva-Voce. The end semester evaluation of the project will be based on the project report, the presentation of the project work undertaken by the student and Viva-Voce.

The weightages for CE and ESE of the project also shall be in the ratio 40:60.

EVALUATION OF PROJECT WORK

CE of the project work shall be done by a departmental committee constituted by the HOD. The committee should consist of a minimum of two faculty members, including the guide.

The assessment is based on presentation, interim report and viva voce. Each internal presentation shall be evaluated based on the following components:

Components	% of Weightage
Understanding of the problem /Concepts	20
Adhering to methodology	15
Quality of presentation and demonstration	15
Quantum of work / effort	25
Organization and content of Project report	5
Viva based on Project	20

Table 4: Components for Continuous Evaluation and the Corresponding Weightage (for Project Work)

End Semester Evaluation (ESE): A board of two examiners appointed by the university shall conduct ESE. The evaluation shall be based on the report, presentation of the work, demonstration of the work and a detailed viva voce based on the work carried out. A candidate will not be permitted to attend the project evaluation without project reports that are duly certified by the guide and HOD. Also, a project will be evaluated only if the candidate attends the ESE presentation and Viva voce on the scheduled date and time. A board shall evaluate a maximum of 10 candidates in a day. The ESE shall consist of the following components:

COMPONENTS	% Weightage
Understanding of the problem/requirements/ concepts related to the project	15
Adhering to methodology (Software engineering phases or research methodology) and the candidates understanding of the components of methodology	15
Quality of Modelling of the problem and solution/ database design / form design / reports / testing (For research projects - relevance / novelty of the work(s) / use of data/ proposal of new models /analysis of algorithms/ comparison and analysis of results / findings)	20
Quality of presentation / demonstration	15
Quantum of work / effort - assessed through the content of report, presentation and viva	25
Organization and content of report	10

Table 5: Components for ESE with the Corresponding Weightage (for Project Work)

Guideline for Preparing Project Report (Both Minor Project Work and Project Work)

i) Arrangement of contents:

The sequence in which the project report material should be arranged and bound should be as follows:

1) Cover Page & Title Page

2) Plagiarism Report

3) Bonafide Certificate

4) Abstract

5) Table of Contents

6) List of Tables

7) List of Figures

8) List of Symbols, Abbreviations and Nomenclature

9) Chapters

10) Conclusion

11) Publications based on the project work (if any)11) Appendices

12) References

The chapters may be broadly divided into 3 parts: (i) introductory chapter, (ii) chapters developing the main theme of the project work, (iii) implementation details (if any) and conclusion. The main text will be divided into several chapters and each chapter may be further divided into several divisions and subdivisions. Each chapter should be given an appropriate title.

Tables and figures in a chapter should be placed in the immediate vicinity of the reference where they are cited. The tables and figures shall be introduced at appropriate places.

Footnotes should be used sparingly. They should be typed single space and placed directly underneath in the very same page, which refers to the material they annotate.

ii) Page Dimension and Binding Specifications: The dimension of the project report should be in A4 size. The project report should be bound using a flexible cover of the thick white art paper. The cover should be printed in black letters and the text for printing should be identical. All the project reports submitted by the students should be plagiarism checked using software and the plagiarism report generated by the software should be verified and signed by the HOD.

MSc Computer Science

LEARNING OUTCOME BASED CURRICULUM FRAME WORK &PROGRAMME STRUCTURE

Graduate Attributes

GRADUATE ATTRIBUTE 1 - SCHOLARSHIP

KANNUR UNIVERSITY graduates will be able inquire critically into their area of study, while being aware of changing state of knowledge both in their own chosen discipline as well as related disciplines

KANNUR UNIVERSITY graduates will have the ability to actively engage in the generation of innovative and relevant knowledge and understanding through inquiry, critique and synthesis going beyond their discipline of specialization.

GRADUATE ATTRIBUTE 2 - CRITICAL CITIZENSHIP AND THE SOCIAL GOOD

KANNUR UNIVERSITY graduates will be engaged, committed and accountable agents of social good. They must aspire to contribute to social justice and environmental sustainability, appreciative of the complexity of historical contexts and societal conditions through their roles as professionals and members of local and global communities.

KANNUR UNIVERSITY graduates will be committed to furthering gender and social equality and empathetically engage with all forms of difference including, conflicting intellectual traditions, religious and cultural practices, language, region and nationality.

GRADUATE ATTRIBUTE 3 - LIFELONG LEARNING

KANNUR UNIVERSITY graduates will be Lifelong Learners, committed to and capable of continuous collaborative and individual learning and critical reflection for the purpose of furthering their understanding of the world and their place in it.

Programme Outcomes (PO)

SL #	Outcome
PO1	**Critical Thinking**: Take informed actions after identifying the assumptions that frame our thinking and actions, checking out the degree to which these assumptions are accurate and valid, and looking at our ideas and decisions (intellectual, organizational, and personal) from different perspectives
PO2	**Problem Solving**: Identify, formulate, conduct investigations, and find solutions to problems based on in-depth knowledge of relevant domains
PO3	**Communication**: Speak, read, write and listen clearly in person and through electronic media in English/language of the discipline, and make meaning of the world by connecting people, ideas, books, media and technology.
PO4	**Responsible Citizenship**: Demonstrate empathetic social concern, and the ability to act with an informed awareness of issues
PO5	**Ethics**: Recognize different value systems including your own, understand the moral dimensions of your decisions, and accept responsibility for them.
PO6	**Self-directed and Life-long Learning**: Acquire the ability to engage in independent and life-long learning in the broadest context socio- technological changes
PO7	**Environmental Sustainability and Global Perspective: -** Develop an understanding of global standards to foster legal environment. Learn and practice to critically analyse the legal issues from local, national and international concerns

Programme Specific Outcomes (PSO)

SL #	Outcome
PSO1	Familiar with the entrenched concepts of Computer Science and Applications.
PSO2	Attain skills to design Algorithms and Programs.
PSO3	Design, build up, put into practice and test software systems to meet the given specifications.
PSO4	Enhance the knowledge about emerging topics in Computer Science
PSO5	Attain prerequisite skills to address research problems in computer science.
PSO6	Furnish the candidate to do the real time jobs linked with Information Technology and Computer Application.

Mapping of PSOs to POs

	PO1	PO2	PO3	PO4	PO5	PO6	PO7
PSO1	✓	✓	✓			✓	✓
PSO2	✓	✓				✓	✓
PSO3	✓	✓	✓	✓	✓	✓	✓
PSO4	✓	✓	✓	✓	✓	✓	✓
PSO5	✓	✓	✓	✓	✓	✓	✓
PSO6	✓	✓	✓	✓	✓	✓	✓

MSc. Computer Science

Programme Structure

Course Structure

Distribution of Credits for the MSc. Computer Science Programmes with effect from 2023- 24 Onwards

	1	2	3	4	5	6	7	8	Total Credits
	Discipline Specific		**Electives**						
Sem	**Core (DSC)**	**Elective (DSE)**	**Interdisciplinary / Multidisciplinary / Open**	**Ability Enhancement Course (AEC) 2 Credits**	**Skill Enhancement Course (SEC) 2 Credits**	**Value Added Course / MOOC 2 Credits**	**Internship / Field Visit / Minor Project / Institutional Industrial Visit 2 Credits**	**Dissertation / Major Project**	

1	MSCSC01DSC01(3C)
	MSCSC01DSC02(4C)
	MSCSC01DSC03(3C)
	MSCSC01DSC04(4C)
	MSCSC01DSC05(2C)
	MSCSC01DSC06(2C)
		MSCSC01DSE01
		MSCSC01DSE02
		MSCSC01DSE03							

	18 Credits	**3 Credits each**							**21**

2	MSCSC02DSC07(3C)
	MSCSC02DSC08(3C)
	MSCSC02DSC09(3C)
	MSCSC02DSC10(3C)
	MSCSC02DSC11(3C)
	MSCSC02DSC12(2C)
	MSCSC02DSC13(2C)
				MCCSA02AECXX (2C)	MCCSA02SECXX (2C)				

	19 Credits			**4 Credits from Any of These**					**23**

3	MSCSC03DSC14(4C)
	MSCSC03DSC15(4C)
	MSCSC03DSC16(3C)
	MSCSC03DSC17(2C)
		MSCSC03DSE04
		MSCSC03DSE05
		MSCSC03DSE06
		MSCSC03DSE07
		MSCSC03DSE08
		MSCSC03DSE09
			MSCSC03MDC01 to 7 (Offered for Other Department			MSCSC03VACXX	MSCSC03DSC18		

	13Credits	**3 Credits Each**	**4 Credits Each**			**2 Credits***	**2 Credits**		**22**
4		Elective III Elective IV (MSCSC04DSE10 to 25) (4C) Elective V MSCSC04DSE26(2C)						MSCSC04DSC19 (8C)	
		10 Credits from Any of These						**8 Credits**	**18**

*MOOC Credit will not be counted for CGPA. However, it is compulsory. Credit earned by the students for the MOOC will be entered as provided by the MOOC provider.

LEGEND	
Item	**Description**
C	Credits
E	External Component (Marks)
ESA	End Semester Evaluation
CE	Continuous Evaluation
P	Practical Hours
T	Total
Tt	Tutorial
CS	Case Study
VAC	Value Added Course
SEC	Skill Enhancement Course
AEC	Ability Enhancement Course
TEC	Technology Enhancement Course
DSC	Department Specific Core
DSE	Department Specific Elective
MDC	Multidisciplinary Elective Course

Semester I

No	Course Code	Course Name	C	Hrs./wk.			Assessment Weightage (%)		
				L	P	Tt	ESA	CE	T
1.1	MSCSC01DSC01	Mathematical Foundations for Computer Science	3	3	0	1	60	40	100
1.2	MSCSC01DSC02	System Software and Operating Systems	4	4	0	1	60	40	100
1.3	MSCSC01DSC03	Time Series Analysis and Forecasting	3	3	0	1	60	40	100
1.4	MSCSC01DSC04	Database Management System	4	4	0	1	60	40	100
1.5	MSCSC01DSEXX	Elective I-DSE (POOL A)	3	3	0	1	60	40	100
1.6	MSCSC01DSC05	Lab I: (i)DBMS (ii) SSOS	2	0	4	0	60	40	100
1.7	MSCSC01DSC06	Lab II: Principles of Programming and Numerical Methods	2	0	4	0	60	40	100
		Total	**21**	**17**	**8**	**5**			

S1 - List of Electives for DSE (POOL A)		
No	**Course Code**	**Course Name**
1	MSCSC01DSE01	Principles of Programming and Numerical Methods using Python
2	MSCSC01DSE02	Principles of Programming and Numerical Methods using C
3	MSCSC01DSE03	Principles of Programming and Numerical Methods using C++

Semester II

No	Course Code	Course Name	C	Hrs./wk.			Assessment Weightage (%)		
				L	P	Tt	ESA	CE	T
2.1	MSCSC02DSC07	Algorithms and Data Structure	3	3	0	1	60	40	100
2.2	MSCSC02DSC08	Digital Signal Processing	3	3	0	1	60	40	100
2.3	MSCSC02DSC09	Artificial Intelligence	3	3	0	1	60	40	100
2.4	MSCSC02DSC10	Theory of Computation	3	3	0	1	60	40	100
2.5	MSCSC02DSC11	Computer Network and Linux Administration	3	3	0	1	60	40	100
2.6	MSCSC02DSC12	Lab III: Algorithms and Data Structure	2	0	4	0	60	40	100
2.7	MSCSC02DSC13	Lab IV: (a)Digital Signal Processing (b)Time Series Analysis and Forecasting	2	0	4	0	60	40	100
2.8	Offered by other departments	SEC/AEC	2	2	0	0	60	40	100
2.9	Offered by other departments	SEC/AEC	2	2	0	0	60	40	100
		Total	**23**	**19**	**8**	**5**			

AECs (offered for other Departments)

No	Course Code	Course Name	C	T	P	Tt
1	MCCSA02AEC01	Algorithm Analysis	2	2	0	2
2	MCCSA02AEC02	Artificial Intelligence and Daily Life	2	2	0	2
3	MCCSA02AEC03	Python Programming	2	2	0	2
4	MCCSA02AEC04	Data Processing with Python	2	2	0	2
5	MCCSA02AEC05	Fundamentals of Big Data	2	2	0	2
6	MCCSA02AEC06	Fundamentals of Programming Methodology	2	2	0	2
	Total		**2**	**2**	**0**	**2**

SECs (offered for other Departments)

No	Course Code	Course Name	C	T	P	Tt
1	MCCSA02SEC01	Basics of Machine Learning	2	2	0	2
2	MCCSA02SEC02	Data Science Fundamentals	2	2	0	2
3	MCCSA02SEC03	Optimization Techniques	2	2	0	2
4	MCCSA02SEC04	Scientific Computing	2	2	0	2
5	MCCSA02SEC05	The Art of E - Documentation using Latex	2	2	0	2

6	MCCSA02SEC06	Fundamentals of Digital Skilling using Google Workspace for Education	2	2	0	2
7	MCCSA02SEC07	Image Processing using Python	2	2	0	0
8	MCCSA02SEC08	Fundamentals of Electrical and Electronics Engineering	2	2	0	0
9	MCCSA02SEC09	Data Analysis and Plotting	2	2	0	0
10	MCCSA02SEC10	Quantum computing	2	2	0	0
	Total		**2**	**2**	**0**	**2**

Semester III

No	Course Code	Course Name	C	Hrs./wk.			Assessment Weightage (%)		
				L	P	Tt	ESA	CE	T
3.1	MSCSC03DSC14	Machine Learning Techniques	4	4	0	1	60	40	100
3.2	MSCSC03DSC15	Computer Graphics and Image Processing	4	4	0	1	60	40	100
3.3	MSCSC03DSC16	Quantum Computing	3	3	0	1	60	40	100
3.4	Offered by other departments	IDC/MDC	4	4	0	0	60	40	100
3.5	MSCSC03DSEXX	Elective II -DSE (POOL B)	3	3	0	0	60	40	100
3.6	MSCSC03DSC17	Lab V: (i)Machine Learning (ii) CG and IP	2	0	4	1	60	40	100
3.7	MSCSC03DSC18	Mini Project	2	0	3	1	60	40	100
3.8	MSCSC03VAC01	MOOC	2*	0	0	0			**
		Total	**22**	**18**	**7**	**5**			

*MOOC Credit will not be counted for CGPA. However it is compulsory.

** Marks earned by the students for the MOOC will be entered as provided by the MOOC provider.

S3 - List of Electives for DSE (POOL B)		
No	**Course Code**	**Course Name**
1	MSCSC03DSE04	Foundations of Data Science
2	MSCSC03DSE05	Foundations of Natural Language Processing
3	MSCSC03DSE06	Speech Audio and Video Forensics
4	MSCSC03DSE07	Internet of Things
5	MSCSC03DSE08	Pattern Recognition
6	MSCSC03DSE09	Computer Vision

Multi-Disciplinary Courses (offered for other Departments)						
No	**Course Code**	**Course Name**	**C**	**T**	**P**	**Tt**
1	MSCSC03MDC01	Design and Analysis of Algorithms	4	4	0	0
2	MSCSC03MDC02	Principle of Programming and Numerical Methods	4	4	0	0
3	MSCSC03MDC03	Java Programming	4	4	0	0
4	MSCSC03MDC04	Machine Learning	4	4	0	0
5	MSCSC03MDC05	Foundations in Data science	4	4	0	0
6	MSCSC03MDC06	Digital Signal Processing	4	4	0	0
7	MSCSC03MDC07	Quantum Computing and Information Theory	4	4	0	0
	Total		**4**	**4**	**0**	**0**

Semester IV

No	Course Code	Course Name	C	Hrs./wk.			Assessment Weightage (%)		
				L	P	Tt	ESA	CE	T
4.1	MSCSC04DSEXX	Elective III-DSE (POOL C)	4	4	0	0	60	40	100
4.2	MSCSC04DSEXX	Elective IV- DSE(POOL C)	4	4	0	0	60	40	100
4.3	MSCSC04DSE26	Elective V*	2	2	0	0	60	40	100
4.4	MSCSC04DSC19	Project	8	0	15	5	60	40	100
		Total	**18**	**10**	**15**	**5**			

*Elective V is meant to foster the students with tools and technologies that they need to know and make use in the design and development of software applications. Seminar Report/ Case study Report of a specific technology should be submitted by each student for the evaluation. The mode of evaluation of this course shall be based on the presentation, report and viva.

S4 - List of Discipline Specific Electives (DSE III/DSE IV) (POOL C)

No	Course Code	Type of Elective	Course Name
1	MSCSC04DSE10	DSE	Software Engineering
2	MSCSC04DSE11	DSE	Cyber Physical Systems
3	MSCSC04DSE12	DSE	Digital Forensics
4	MSCSC04DSE13	DSE	Block chain and Crypto currency technologies
5	MSCSC04DSE14	DSE	Grid and Cloud Computing
6	MSCSC04DSE15	DSE	Embedded System
7	MSCSC04DSE16	DSE	High Performance Computing
8	MSCSC04DSE17	DSE	Data and Information Visualization
9	MSCSC04DSE18	DSE	Information Retrieval System
10	MSCSC04DSE19	DSE	Biometric Image Processing
11	MSCSC04DSE20	DSE	Java Programming
12	MSCSC04DSE21	DSE	Nature Inspired Computing
13	MSCSC04DSE22	DSE	Big Data Analytics
14	MSCSC04DSE23	DSE	Astronomical Image Processing
15	MSCSC04DSE24	DSE	Geographic Information System
16	MSCSC04DSE25	DSE	Operation Research

Semester I

CORE COURSE

MSCSC01DSC01 MATHEMATICAL FOUNDATIONS FOR COMPUTER SCIENCE

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course deals with mathematical concepts like elementary discrete mathematics, probability & statistics and linear algebra.

Course Objectives:

- Impart knowledge on mathematical logic.
- Give basic idea set theory, relations and functions and their problem solving.
- Familiarize measures of central tendency and measures of dispersion.
- Impart knowledge on probability and its distributions.
- Familiarize matrices and its operations, vector space and Eigen vectors.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire knowledge about mathematical logic, set theory and relations
CO2	Use functions, partial ordering and counting techniques to solve problems.
CO3	Understand measures of central tendency, measures of dispersion, probability and its distributions.
CO4	Basic understanding of linear algebra

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓			✓	✓
CO3	✓			✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Mathematical logic: Propositional and Predicate Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Normal Forms

Set and relations: Set Operations, Properties of Relations, Representing relations- matrices & digraphs. Closure of Relations, Composition of relations, Equivalence Relations.

Module 2: Functions: Types of Functions, Composition of Functions and Inverse Functions, Some important functions- floor & ceiling, Recursive functions

Partitions- Partial Ordering, Hasse Diagram, Lattice –Types, Properties, Basics of Counting, Pigeonhole Principle, Permutations and Combinations, Inclusion- Exclusion Principle.

Module 3: Measures of Central Tendency, Measures of Dispersion,Coefficient of Variation, Covariance.

Probability – Random experiment, Sample point, Sample space, Events, Algebra of events, Statistical regularity, Frequency and Classical definitions, Axiomatic approach to probability, Probability Space and probability measure, Addition theorem, Conditional probability, Multiplication theorem, Independence of events, Bayes' theorem and applications.

Discrete Distributions – One point, two point distributions, Uniform, Point binomial, Poisson, Continuous Distributions – Normal, Exponential

Module 4: Matrices and determinants: matrix, types of matrices, operations on matrices, transpose of a matrix, Determinants-properties of determinants- inverse of a matrix- Rank of a Matrix, Trace of a Matrix. Solving Linear Equations using Matrices – Matrix solution, Gauss Elimination Method

Vector Space, Subspace, Linear Dependence and Independence, Basis and Dimension, Linear Transformations, Matrices Related to Linear Transformations, Eigenvalues and Eigenvectors.

Module X (For Additional Reading and Comprehension by the Students):

Mathematical Induction, Recurrence Relation, Generating function, Group Theory: Groups, Subgroups

Discrete Distributions– Geometric, Hyper geometric and Negative binomial distributions

Continuous Distribution - Rectangular, Beta, Gamma, log normal distribution. Consistency of a linear system, Diagonalization of a matrix, Diagonalization of a symmetric matrix.

Core Compulsory Readings

1. Kenneth H. Rosen, Discrete Mathematics and Applications, TMH 2003
2. Elementary Linear Algebra - Devi Prasad (Narosa Pub. House, 2006)
3. Fundamentals of Mathematical Statistics - S. C.Gupta&V.K.Kapoor (Sulthan Chand & Sons)

Core Suggested Readings

1. Discrete Mathematics and Its Applications with Combinatorics and Graph Theory, Kamala Krithivasan, McGraw Hill Education, 2011 (Seventh Edition).
2. J.P.Tremblay and R Manohar, Discrete Mathematical Structures with Applications to Computer Science, TMH 2001
3. Discrete Mathematics, N Ch S N Iyengar, V M Chandrasekharan, KA Venkatesh, PS Arunachalam, Vikas Publishing, 2003.
4. Introduction to Probability and Statistics for Engineers and Scientists- S.M. Ross (Elsever)
5. Linear Algebra - A Geometric Approach - S. Kumaresan (Prentice Hall India)

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Questioning and Answering

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Define tautology and contradiction.
2. Show that for any two sets A−(A∩B) = A−B
3. Explain Equivalence relation.
4. Let R be a partial ordering of the set of all devisors of 64. Construct the hasse diagram for it. Find the meet and join.
5. Three dice are rolled together. What is the probability of getting at least one '4'?
6. Use Gauss elimination to solve the following system of linear equations.

$$2X + Y + Z = 10$$

$$3X + 2Y + 3Z = 18$$

$$X + 4Y + 9Z = 16$$

CORE COURSE

MSCSC01DSC02 SYSTEM SOFTWARE AND OPERATING SYSTEMS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4/1	0	5	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course is to provide students with basic knowledge of system software. This course will cover assemblers, linkers, loaders and compilers. Particular emphasis will be given to major OS subsystems: process management (processes, threads, CPU scheduling, synchronization, and deadlock), memory management (segmentation, paging, swapping), file systems, I/O systems and mass storage structure.

Course Objectives:

- To know the design and implementation of assemblers, macro processor, linker, loader and compiler.
- To explain the main components of OS and their working.
- To familiarize the operations performed by OS as a resource Manager.
- To impart various scheduling policies of OS
- To teach the different memory management techniques.
- To explain file system, mass storage structure and input/output management.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire knowledge about Assembler, Linkers and Loaders.
CO2	Understand Process scheduling, process synchronization and methods to handle deadlocks.
CO3	Understand Memory Management and file management techniques.
CO4	Understand I/O systems, mass storage structure and different disk scheduling algorithms.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓		✓	✓	✓

COURSE CONTENTS

Module 1: Assemblers: Elements of Assembly Language Programming, Overview of Assembly Process, Design of Two pass Assembler, Macros and Macro Processors, Macro definition, call and expansion, Nested Macro calls, Design of Macro pre-processor. Linkers: Linking and Relocation concepts, Design of linkers, Self relocating programs. Loaders: introduction to loaders - functions of loaders- Compilers: Introduction to compilers -Different Phases-Lexical Analysis- role of the lexical analyzer, input buffering, specification of tokens, Recognition of tokens, lexical Analyzer generators, Lex.

Module 2: Introduction to Operating systems**:** Different types of Operating system, Overview of Operating systems, Operating system structures, Process management -Processes, Process Scheduling, Inter Process communication - Communication in client server systems, Threads - Processes Vs Threads, Types of threads, Multicore and Multithreading. CPU Scheduling - Scheduling algorithms. Process synchronization: Critical section Problem, Mutual Exclusion, Requirements, Semaphores, Producer Consumer Problem, Readers Writers Problem, Deadlock :Prevention, Detection and Recovery.

Module 3: Memory Management- Swapping, Contiguous memory allocation, Paging, Segmentation, Segmentation with paging. Virtual memory- Demand paging, processes creation, page replacement, allocation of frames, thrashing. File system interface and Implementation - File concepts, access methods, directory structure, File system implementation, Directory implementation, Allocation methods.

Module 4: I / O Systems - I / O hardware, Application I/O interface, Kernel I / O subsystem, Transforming I / O to hardware operations, STREAMS, Performances. Mass storage structure - Disk structure, Disk scheduling, Disk management, Swap space managements, RAID structure, Disk attachments, Stable storage implementation, Tertiary storage structure.

Module X (For Additional Reading and Comprehension by the Students): Macros-Advanced Macro facilities, Linker-Linking for over-lays, CPU Scheduling- Multiple Processor Scheduling, Algorithm Evaluation- Advanced CPU scheduling. Process synchronization: Monitors, File system interface and Implementation- File system mounting, File sharing, Protection, File system structure, Free space managements, Efficiency and performance, Recovery, Log- structured file system.

Core Compulsory Readings

1. D.M. Dhamdhere, Systems Programming and Operating Systems, TMH, 2003.
2. Silberschatz, A., Galvin, P.B. & Gagne, G. Operating System Concepts, 9th Ed. John Wiley & Sons- India.

Core Suggested Readings

1. Dhamdhere, D. M. Operating Systems, 2nd Ed. The McGraw - Hill Companies.
2. Ditel, Deital and Choffness, Operating Systems, Pearson, 3rdEdn
3. William Stallings, Operating Systems, Internals and Design Principles, 7th Edition Pearson,
4. SibsankarHaldar ,Alex a Aravind, Operating Systems, Pearson Education India, Second impression.
5. Andrew S.Tanenbaum , Albert S.Woodhull, The Minix Book- Operating Systems Design and Implementation, 3rd Edition Pearson(2016).

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Design a two-pass assembler with its necessary phases
2. With an example explain FCFS, SJF and Round Robin CPU Scheduling Algorithms
3. Describe Producer Consumer Problem
4. Illustrate banker's algorithm to avoid deadlocks
5. Describe Paging mechanism with examples
6. Implement any three-page replacement algorithms for the reference string given as follows and find the number of page faults
7. Reference string: 1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5.
8. Explain various File Access methods involved with OS
9. Explain with a diagram a typical PC Bus structure with I/O Devices
10. Apply and illustrate SSTF, FCFS and SCAN disk scheduling algorithms with a request queue of 98, 183, 37, 122, 14, 124, 65, 67 with Head pointer at 53

CORE COURSE

MSCSC01DSC03 TIME SERIES ANALYSIS AND FORECASTING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Time Series Analysis and Forecasting is a practical course that teaches students the essential concepts and techniques for analysing and predicting patterns in time-dependent data. Students will learn how to pre-process and visualize time series data, identify trends and seasonality, and select appropriate models for analysis. The course emphasizes hands-on experience with popular statistical software packages and real-world case studies to reinforce learning. By the end of the course, students will be equipped with the skills to effectively analyse time series data and make accurate forecasts, enabling them to make informed decisions in diverse fields such as finance, economics, and marketing.

Course Objectives:

- Understand the fundamental concepts and characteristics of time series data, including autocorrelation, stationarity, trends, and seasonality.
- Learn various techniques for preprocessing and visualizing time series data to uncover patterns and insights.
- Develop proficiency in selecting and implementing appropriate time series models, such as ARIMA, exponential smoothing, and state space models.
- Gain hands-on experience in forecasting by applying different techniques and evaluating the performance of forecasting models.
- Apply time series analysis and forecasting skills to real-world case studies and projects, enabling students to make data-driven decisions and predictions in practical scenarios.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Develop comprehensive understanding of probability concept.
CO2	Perform Data Pre-processing Techniques, Derive the properties of ARIMA and state-space models.
CO3	Choose an appropriate ARIMA model for a given set of data and fit the model using an appropriate package.
CO4	Compute forecasts for a variety of linear methods and models

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Random variable – Continuous, Discrete, Identically Independent Distribution (IID), Introduction to probability, Conditional probability, Bayes theorem, Probability Distribution, Probability Density function, Expectation value, Moment of probability density function, variance, covariance, standard deviation, auto covariance, central limit theorem, correlation, autocorrelation, partial autocorrelation, random process, random walk, Difference between time series and regression, Difference between probability and likelihood

Module 2: Fourth paradigm of science, Introduction to time series data, characteristics of time series data, Understanding the components of a time series: trend, seasonality, cyclicality, and residual, Stationary and non-stationary Time Series, Vector Valued and Multidimensional Series.Introduction to Data Acquisition and Pre-processing, Data Cleaning – outlier detection and treatment, handling missing data, Smoothing, Detrending, Data Normalization, Data Transformation, Data Integration, Feature Selection, Handling imbalanced data

Module 3: Exploratory data analysis, Explore patterns and trends in time series data, Seasonality analysis and detection methods, Autocorrelation and partial autocorrelation analysis. Trend analysis and trend removal techniques, Seasonal decomposition and visualization of components. Classical Regression in the Time Series Context

Module 4: Introduction to ARMA models, Model identification: selecting appropriate orders of ARIMA models, Parameter estimation and model fitting, Model diagnostics and evaluation: residual analysis, Introduction to SARIMA model. Time series forecasting using ARIMA and SARIMA models.

Module X (For Additional Reading and Comprehension by the Students): Recurrent Neural Networks (RNNs) - Introduction to RNNs, Understanding the recurrent layer,Long Short-Term Memory (LSTM) networks, Gated Recurrent Unit (GRU) networks, Advantagesand use cases

Core Compulsory Readings

1. Rob J Hyndman (2014), Forecasting: Principle & Practice, University of Western Australia
2. R. H. Shumway and D. S. Sto_er (2017), Time Series Analysis and Its Applications (With R Examples, fourth Edition).Springer, New York.

Core Suggested Readings

1. Nonlinear time series analysis .Ruey S. Tsay and Ronngchen. Wiley 2019
2. Enders W. Applied Econometric Time Series. John Wiley & Sons, Inc., 1995.
3. Mills, T.C. The Econometric Modelling of Financial Time Series. Cambridge University Press, 1999
4. Andrew C. Harvey. Time Series Models.Harvesterwheatsheaf, 1993.
5. Andrew C. Harvey. The Econometric Analysis of Time Series. Philip Allan, 1990.

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What are the key characteristics of time series data, and why is it important to understand them in the context of analysis and forecasting?
2. How can you pre-process time series data to handle missing values and outliers effectively?
3. What are the different methods available for visualizing time series data, and how can they help in identifying trends and seasonality?
4. Explain the concept of autocorrelation and its significance in time series analysis.
5. What is stationarity in the context of time series data, and why is it important for modelling and forecasting?
6. Compare and contrast the ARIMA and exponential smoothing models in terms of their assumptions and applicability.
7. How can you evaluate the performance of a forecasting model, and what are some commonly used metrics for this purpose?
8. Describe the process of model selection in time series analysis and the factors to consider when choosing an appropriate model.
9. What are the steps involved in building a forecast using the chosen time series model, and how can you interpret the results?
10. Provide examples of real-world applications of time series analysis and forecasting in different industries or domains.

CORE COURSE

MSCSC01DSC04 DATABASE MANAGEMENT SYSTEM

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4/1	0	5	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The Database Management Systems (DBMS) course is designed to provide a comprehensive understanding of the principles and practices of managing databases. Also focuses on the theoretical foundations and practical applications of DBMS, which play a crucial role in modern information systems.

Course Objectives:

- To introduce students to the basic concepts and the principles of Database Management Systems.
- To understand the value of data integrity, security and privacy concepts.
- To explore normalization, transaction management, indexing and concurrency control.
- To provide a hands-on experience in designing, implementation and management of databases.
- To introduce emerging technologies in DBMS field.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Understand the concept of Database Management Systems
CO2	Understand the concept of transaction management with respect to database.
CO3	Compare different types of NoSQL Databases and their applications
CO4	Demonstrate the detailed architecture and performance tune of Document-oriented NoSQL databases.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓		✓	✓	✓	✓
CO2	✓		✓	✓	✓	✓
CO3			✓	✓	✓	✓
CO4			✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Database Management Concepts: The ER and EER. Relational Algebra and Relational calculus. Functional dependencies -1st, 2nd, 3rd, 4th, BCNF, 5th Normal form. Sub-queries: correlated sub-queries, join, Exist, Any, All, joined relations. Integrity and security: domain constraints, Referential integrity, assertion, triggers. Views: Introduction to views, data independence, security, updates on views, comparison between tables and views.

Module 2: Transaction processing- desirable properties of transaction. Transactions and Schedules – Characterising Schedules based on Recoverability, Serializability of schedules. Concurrency Control in databases: Locking Techniques-Time stamp ordering, Multi version concurrency Control – Granularity of data items.

Module 3: Overview and History of NoSQL Databases. Definition of the Four Types of NoSQL Databases, The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, Impedance Mismatch, The Emergence of NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Application and Integration Databases, Attack of the Clusters.

Module 4: Map-Reduce: Basic Map-Reduce, Partitioning and Combining, Composing Map-Reduce Calculations, A Two Stage Map-Reduce Example, Incremental Map-Reduce Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets.

Module X: (For additional reading and comprehension by the students)

NoSQL Key/Value databases using MongoDB, Document Databases, Document oriented Database Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure.

Core Compulsory Readings

1. HSilbersehatz, Korth and Sudarshan, Database system concepts, 6th edition MGH 2011
2. Ramakrishnan and Gehrke, Database Management Systems, 3rd Edn, Mc Graw Hill, 2003
3. Elmasri and Navathe, Fundementals of Database systems, 5th Edition, Pearson 2009
4. C.J.Date-A.Kannan, S.Swamynathan "An introduction to Database System" 8th Edition, Pearson education O'Reilly, Practical PostgreSQL Shroff Publishers(SPD) 2002.
5. Sadalage, P. & Fowler, M. (2012). NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. (1st Ed.). Upper Saddle River, NJ: Pearson Education, Inc. ISBN- 13: 978-0321826626 ISBN-10: 0321826620
6. Redmond, E. & Wilson, J. (2012). Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement (1st Ed.). Raleigh, NC: The Pragmatic Programmers, LLC. ISBN-13: 978-1934356920 ISBN-10: 1934356921

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes

1. Explain the concept of normalization in the context of database design. Why is it important? Give an example.

2. With corresponding examples, explain the difference between a primary key and a foreign key in a relational database. How it can be represented in a table.
3. Discuss the concept of ACID properties in transaction management. How do they ensure data integrity and consistency.
4. Write the advantages and disadvantages of various database indexes.
5. Write an SQL query to retrieve all students who have got more than 75 percentage of mark in the department of science from the student table.
6. Explain the purpose of a trigger in database. Give an example scenario where a trigger can be useful.
7. Discuss how database concurrency control used to ensure the consistency of data in a multi-user environment.
8. Discuss the role of data warehousing in decision support systems. What are the common challenges to be faces while implementing a data warehouse.
9. Compare and contrast distributed database system and centralized database system.
10. What are NoSQL databases? How it is different from traditional databases? Discuss a scenario where NoSQL database is more suitable than the traditional database.

ELECTIVE COURSE

MSCSC01DSE01PRINCIPLES OF PROGRAMMING AND NUMERICAL METHODS USING PYTHON

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Numerical computational methods help to solve complex mathematical problems which cannot be solved easily by analytical mathematics by using simple arithmetic operations and which requires the use of an algorithm. The understanding of numerical computing helps to simulate various scientific models. The course will focus on different advanced paradigms and approaches using Python programming.

Course Objectives:

- To introduce basic concepts of python programming language.
- Discuss about Errors and Approximations
- Discuss about the concept of numerical computational methods
- Discuss the numerical integration and differentiation

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Acquainted with Numerical Methods.
CO2	Obtain the knowledge about Numerical Integration, Differentiation and its applications
CO3	Understanding the basic concepts of Python programming.
CO4	Ripen skill in programming

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Numerical Methods: Nature of numerical problems; computer based solutions. Errors and Approximations. Nonlinear equations – Bisection Method, Regular- Falsie Method, Newton- Raphson. System of Linear Equations- Gauss elimination, Gauss Jordan elimination, Triangulation method, Iterative method, Jacobi. Case study by writing algorithms.

Module2: Numerical Integration and Differentiation: Concept of differentiation and Integration. Taylors series and Eulers methods-Simpson's Romberg, Gaussian, Runge- Kutta methods. Case study by writing algorithms.

Module 3: Features of Python, Different Methods to Run Python, Basic Elements (Objects, Expressions, Numerical Types, Strings, Variables), Comments, Indentation in Python, Input and Output in Python, import function, Operators in Python, Branching (if, else, elif), Iteration (while, for), range and enumerate functions, Tuples, Lists, Sets, Dictionaries, Built-in methods of lists, sets and dictionaries, Mutable and Immutable Objects.

Module 4: Functions Definition, Function Calling, Function Arguments (Required, Keyword, Default), Recursion, Modules, Built-in Modules, Creating Modules, File Handling (Opening, Closing, Writing, Reading), Exceptions, Built-in Exceptions (IndexError, OverflowError, ZeroDivisionError, RuntimeError), Exception Handling. Class Definition, Object Creation, Built-in Attribute Methods, Object Oriented Programming Features of Python. Arrays in Python, Numpy Module, ndarray, Creating Arrays (array, zeros, ones, empty, linspace, arrange, random), Two-Dimensional Array, Indexing, Slicing, Iterating, Copying, Splitting, Shape Manipulation (reshape, transpose, resize), Arithmetic Operations on Arrays. Data Visualization in Python matplotlib Module, pyplot, plot(), scatter, bar charts, Formatting, figure(), subplot(), text(), xlabel(), ylabel(), title(), Plotting Simple Mathematical Functions (sin x, x^2).

Core Compulsory Readings

1. Discrete Mathematical Structures with Application to Computer Science-McGraw Hill
2. Introductory Methods of Numerical Analysis – January 2012 PHI, Sastry S.S

Core Suggested Readings

1. Balagurusamy, E., "Numerical Methods", Tata McGraw-Hill, New Delhi, 1999.
2. R.G.Dromey , How to solve it by computer, Pearson education, fifth edition, 2007.
3. Taming Python By Programming, Dr. Jeeva Jose, Khanna Publishing
4. Introduction to Computation and Programming Using Python with Application to Understanding Data - John V. Guttag, PHI (2016)
5. https://www.numpy.org/devdocs/user/quickstart.html
6. https://matplotlib.org/users/pyplot_tutorial.html

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Write python program to demonstrate Bisection method

2. Write the algorithm of False position method

3. Explain Gaussian elimination method with an example

4. Write the algorithm to implement Gauss-Jordan method

5. Write python program for computing the value of nP_r

6. Explain the concept of differentiation and Integration with examples.

7. Find the real root of the equation $x^3-2x-5 = 0$

8. Find the positive root, between 0 and 1, of the equation $x = e^{-x}$ to a tolerance of 0.05%

9. Find a root, correct to three decimal places and laying between 0 and 0.5, of the equation

   $4\ e\text{-}^{x}\ \text{Sin}\ x - 1 = 0$

10. Obtain the solution of the following system using the Jacobi iteration method

    $2x_1 + x_2 + x_3 = 5$

    $3x_1 + 5x_2 + 2x_3 = 15$

    $2x_1 + x_2 + 4x_3 = 8$

11. Solve the system

    $2x_1 + 4x_2 - 6x_3 = -8$

    $x_1 + 3x_2 + x_3 = 10$

    $2x_1 - 4x_2 - 2x_3 = -12$

    using Gauss- Jordan method

ELECTIVE COURSE

MSCSC01DSE02PRINCIPLES OF PROGRAMMING AND NUMERICAL METHODS USING C

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Numerical computational methods help to solve complex mathematical problems which cannot be solved easily by analytical mathematics by using simple arithmetic operations and which requires the use of an algorithm. The understanding of numerical computing helps to simulate various scientific models. The course will focus on different paradigms and approaces using C programming.

Course Objectives:

- To introduce basic concepts of Procedure Oriented Programming.
- Discuss about Errors and Approximations
- Discuss about the concept of numerical computational methods
- Discuss the numerical integration and differentiation

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquainted with Numerical Methods.
CO2	Obtain the knowledge about Numerical Integration and Differentiation
CO3	Understanding the basic concepts of C programming.
CO4	Develop programming skill

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Numerical Methods: Nature of numerical problems; computer based solutions. Errors and Approximations. Nonlinear equations – Bisection Method, Regular- Falsie Method, Newton- Raphson. System of Linear Equations- Gauss elimination, Gauss Jordan elimination, Triangulation method, Iterative method, Jacobi. Case study by writing algorithms.

Module 2: Numerical Integration and Differentiation: Concept of differentiation and Integration. Taylors series and Eulers methods-Simpson's Romberg, Gaussian, Runge- Kutta methods. Case study by writing algorithms.

Module 3: Algorithms and Flow charts: Definitions, Symbols, Program : structure, top- down design, source code, object code, executable file, file extensions. Importance of C; Basic structure of C, Programming style, executing a c program. Character set, C tokens, Keywords, identifiers, Constants, data types, declaration of variables, arithmetic operators , logical operators, Relational operators, Assignment operators, Increment and decrement operators, conditional operators, Bitwise operators. Precedence and order of evaluation. type conversion in expression. common programming errors, program testing and debugging, program efficiency. Managing Input output operation: reading a character, writing a character, formatted input output. Branching statements-if, if..else, nested if...else, else...if ladder, switch statement, go to statement. Looping statements- while, do...while, for loop. Break and continue statements.

Arrays: One dimensional arrays, two dimensional arrays, Initializing array elements, Multidimensional arrays. Strings: declaration and initializing, reading and writing. Arithmetic operations on character. String handling functions Functions: Library and user defined, defining a function, calling a function. Parameter passing techniques, Scope and life time of variables in function, recursive functions, arrays and functions.

Module 4: Structure and union: definition, giving values to members, initialization. Array of structures, array with in structure, structure with in structure, union. Pointers: accessing the address of a variable, declaration and initializing pointers, accessing a variable through its pointers, pointer arithmetic, pointers and arrays (pointer to array and array of pointers) , pointers and character string , pointer and functions. Dynamic memory allocation: malloc(), calloc(), free(), realloc().

File Management: Text and binary files, Defining and opening a file, closing a file, input and output operations on file, error handling, random access file. Command line arguments.

Core Compulsory Readings

3. Discrete Mathematical Structures with Application to Computer Science-McGraw Hill

4. Introductory Methods of Numerical Analysis – January 2012 PHI, Sastry S.S

Core Suggested Readings

1. Balagurusamy, E., "Numerical Methods", Tata McGraw-Hill, New Delhi, 1999.

2. Computer Basics and c Programming, V. Rajaraman, PHI,2008

3. Let us C, YeshvanthKanethkar, 3rdEdn,BPB

4. C by Example, Noel Kalicharan, Cambridge Universitypress.

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Write C programme to demonstrate Bisection method
2. Write the algorithm of False position method

3. Explain Gaussian elimination method with an example

4. Write the algorithm to implement Gauss-Jordan method

5. Write C program for computing the value of nP_r

6. Explain the concept of differentiation and Integration with examples.

7. Find the real root of the equation $x^3-2x-5 = 0$

8. Find the positive root, between 0 and 1, of the equation $x = e^{-x}$ to a tolerance of 0.05%

9. Find a root, correct to three decimal places and laying between 0 and 0.5, of the equation

$4 e^{-x} \text{Sin } x - 1 = 0$

10. Obtain the solution of the following system using the Jacobi iteration method

$2x_1 + x_2 + x_3 = 5$

$3x_1 + 5x_2 + 2x_3 = 15$

$2x_1 + x_2 + 4x_3 = 8$

11. Solve the system

$2x_1 + 4x_2 - 6x_3 = -8$

$x_1 + 3x_2 + x_3 = 10$

$2x_1 - 4x_2 - 2x_3 = -12$

using Gauss- Jordan method

11. Write the algorithm for simulating Runge- Kutta method.

ELECTIVE COURSE

MSCSC01DSE03PRINCIPLES OF PROGRAMMING AND NUMERICAL METHODS USING C++

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Numerical computational methods help to solve complex mathematical problems which cannot be solved easily by analytical mathematics by using simple arithmetic operations and which requires the use of an algorithm. The understanding of numerical computing helps to simulate various scientific models. The course will focus on different advanced paradigms and approaches using C++ programming.

Course Objectives:

- To introduce basic concepts of OOP
- Discuss about Errors and Approximations
- Discuss about the concept of numerical computational methods
- Discuss the numerical integration and differentiation

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquainted with Numerical Methods.
CO2	Obtain the knowledge about Numerical Integration and Differentiation
CO3	Understanding the basic concepts of C++ programming.
CO4	Develop skill in programming

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Numerical Methods: Nature of numerical problems; computer based solutions. Errors and Approximations. Nonlinear equations – Bisection Method, Regular- Falsie Method, Newton- Raphson. System of Linear Equations- Gauss elimination, Gauss Jordan elimination, Triangulation method, Iterative method, Jacobi. Case study by writing algorithms.

Module 2: Numerical Integration and Differentiation: Concept of differentiation and Integration. Taylors series and Eulers methods-Simpson's Romberg, Gaussian, Runge- Kutta methods. Case study by writing algorithms.

Module 3: Principles of object oriented programming; OOP paradigm; Basic concepts of OOP; Benefits; applications. Introduction to C++, Structure of C++ program; Tokens, Keywords, identifiers and constants; Data types, symbolic constants; type compatibility; declaration and dynamic initialization of variables; reference variables.Operators, manipulators; type cast operators; Expressions, implicit conversions; operator overloading; operator precedence; Control structures.

Functions; function overloading; friend and virtual functions; Math library functions. Structures; Specifying a class; Defining member functions; making an outside function inline; nesting of member functions; private member functions; arrays within a class; memory allocation for objects; static data members; static member functions; arrays of objects; objects as function arguments; friendly functions; returning objects; const member functions; pointer to members; Local classes.

Module 4: Constructors and destructors; dynamic initialization of objects; copy constructor; Dynamic constructors; const objects; Destructors. Operator overloading – definition; overloading unary operators; overloading binary operators; overloading binary operators using friends; manipulation of strings using operators; rules for overloading operators. Type conversions.

Inheritance – defining derived classes; making a private member inheritance; Types of inheritance; virtual base classes; abstract classes; constructors in derived classes; Nesting of classes. Pointers; Pointers to objects; Pointers to derived classes; virtual functions; pure virtual functions.

C++ streams; stream classes; unformatted I/O operations; Formatted console I/O operations; Managing output with manipulators. Files – classes for file stream operations; Opening and closing a file; file modes; file pointers and their manipulations; Sequential input and output operation.

Core Compulsory Readings

5. Discrete Mathematical Structures with Application to Computer Science-McGraw Hill

6. Introductory Methods of Numerical Analysis – January 2012 PHI, Sastry S.S

7. Stroustrup, Bjarne (2000). Programming Principles and Practices using C++ (2nd

   Edition). Addison-Wesley. 2014

Core Suggested Readings

1. Balagurusamy, E., "Numerical Methods", Tata McGraw-Hill, New Delhi, 1999.

2. Balagurusamy, Object Oriented Prog With C++,5e, McGraw-Hill Education (India) Pvt Limited, 2011

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Write C++ programme to demonstrate Bisection method
2. Write the algorithm of False position method
3. Explain Gaussian elimination method with an example
4. Write the algorithm to implement Gauss-Jordan method
5. Write C++ program for computing the value of nP_r
6. Explain the concept of differentiation and Integration with examples.
7. Find the real root of the equation $x^3-2x-5 = 0$
8. Find the positive root, between 0 and 1, of the equation $x = e^{-x}$ to a tolerance of 0.05%
9. Find a root, correct to three decimal places and laying between 0 and 0.5, of the equation $4 e^{-x} \sin x - 1 = 0$
10. Obtain the solution of the following system using the Jacobi iteration method

$$2x_1 + x_2 + x_3 = 5$$

$$3x_1 + 5x_2 + 2x_3 = 15$$

$$2x_1 + x_2 + 4x_3 = 8$$

11. Solve the system using Gauss- Jordan method

$$2x_1 + 4x_2 - 6x_3 = -8$$

$$x_1 + 3x_2 + x_3 = 10$$

$$2x_1 - 4x_2 - 2x_3 = -12$$

12. Write the algorithm for simulating Runge- Kutta method.

END OF SEMESTER 1

Semester II

CORE COURSE

MSCSC02DSC07 ALGORITHMS AND DATA STRUCTURE

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Course Description:

This is the one of the basic courses in computer science deals the representation and storage of data in a most efficient and effective ways. This course primary objective is to study and analyzes different algorithms and its complexities in manipulation and storage of data in computer in a most effective way.

Course Objectives:

- To impart knowledge about various data structures, their representation, and applications.
- To acquaint with various Algorithmic techniques and applications.
- To familiarize the design and analysis of algorithms
- To familiarize the usage of linear and non-linear data structures and its implementation.
- Acquire the knowledge of various searching and sorting techniques.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Understand the basic concepts of different algorithms.
CO2	Acquire knowledge on linear data structures
CO3	Acquire knowledge on non-linear data structures
CO4	Understand different searching sorting techniques

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓		✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Algorithms: Definition, Pseudocode Representation – Algorithm analysis: time and space complexity. Growth functions: asymptotic notations, cost estimation based on key operations- Big Oh, Big Omega, Little Oh, Little Omega and Theta notations. Analyzing algorithm control structures, Solving recurrences: Iteration method, Substitution method, Recursion Tree method. Master's theorem- 3 cases. Basic technique for design of efficient algorithm: Brute Force approach, Divide-and-Conquer approach, Branch-and-Bound technique. Greedy approach, Dynamic programming, Backtracking.

Module 2: Data structures: Definition and classification. Linear data structure: Array- operations, polynomial representation with arrays; Stack: operations on stack. Application of stack – concept of recursion, types of recursions - Prefix and postfix expression- conversion of infix to postfix expression – postfix expression evaluation. Queues: operation on queue. Circular queue, dequeue and priority queue. Application of queue: job scheduling. Linked list: single linked list-structure and implementation; operations – traversing, add new node, delete node, reverse a list, search and merge two singly linked lists. Circular linked list– advantage. Queue as circular linked list. Doubly linked list, operations – add/delete nodes, advantages.

Module 3: Non- linear data structure: Tree- basic terminologies and properties; representation of binary tree, operations on binary tree; type of binary tree, forest. Tree traversal: in order, pre order and post order traversals. Binary search tree. Application of tree, AVL tree, Graphs: basic terminologies, representation of graph, matrix representation of graphs. Graph traversals - depth-first traversal – breadth-first traversal - applications of graphs – minimum spanning tree – Prim's and Kruskal's algorithms- shortest-path algorithm – Dijkstra's algorithm

Unit 4: Searching and sorting techniques : Linear search- Sentinel Linear search- Ternary search- Jump search Binary search - Hash table, . Sorting Techniques: Insertion sort, Bubble sort, Selection sort, Quick sort and Merge sort. Comparison of sorting algorithms. Case Study: Searching Algorithms for Big Data

Module X (For Additional Reading and Comprehension by the Students):

Analysis of Strassen algorithm for matrix multiplication, Complexity Classes: P, NP, NP Hard and NP Complete problems. B-Tree, B+ Tree, Red-black Tree, Huffman algorithm. Representations and operations of sets- Bigdata searching techniques

Core Compulsory Readings

1. Thomas H Cormen, Charles E Leiserson, and Ronald L Rivest, Introduction to Algorithms, 3rd Edition, Prentice Hall of India Private Limited, New Delhi.
2. Sahni and Mehta, Fundamentals of Data Structures in C++, 2nd Edn, University Press
3. Samanta, Classic Data structures, Second Edition, PH
4. Sahni, Rajasekaran, Fundamentals of Algorithms, 2ndEdn, University Press
5. Sahni, Rajasekaran, Fundamentals of Algorithms, 2ndEdn, University Press
6. M. A. Weis, Data Structures and Algorithm Analysis in C++, Pearson Edu. Asia, 2013

Core Suggested Readings

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, Design and Analysis of Computer Algorithms, Addison Wesley.
2. Pallaw, V K, Design and Analysis of Algorithms, Asian Books Private Ltd, 2012
3. Pandey H M, Design and Analysis of Algorithms, University Science Press, 2013
4. OdedGoldreich, P,NP and NP- Completeness, Cambridge University Press, 2011.
5. Donald Knuth, The Art of Computer Programming, Fundamental Algorithms, Volume- 1
6. Anany Leviton, Introduction to the Design and Analysis of Algorithms, 3rd Edition, Pearson Education.

TEACHING LEARNING STRATEGIES

Lecturing, case study/mini projects, Team Learning, presenting seminars on selected topics, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print.

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is data structure?

2. What are the characteristics of an algorithms?

3. Differentiate between time and space complexity of an algorithm?

4.Explain the various techniques for designing an efficient algorithm.

5. Distinguish between linear and non-linear data structure?

6. Explain the different types of arrays and its operations.

7. What is bubble sort?

8. What is binary search?

9. What is recursion? Explain how it is implemented?

10.What is stack? Explain its basic operation.

11. Explain the applications of stack?

12. What is meant by FIFO?

13.What is meant by LIFO?

15. What are priority queues?

16. What are linked list?

17. Explain the different operations on linked list.

18. What is doubly linked list?

19. Explain the applications of queue.

20. What are the different types of lists? Explain the applications of each one.

21. What is BST?

22.Explain quick sort algorithm.

23. What are the different tree traversing algorithms? Explain.

23. Explain selection sort algorithm?

25. Compare and contrast selection sort and insertion algorithm with examples.

CORE COURSE

MSCSC02DSC08 DIGITAL SIGNAL PROCESSING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Digital signal processing" is a core course. Course starts from the basic concepts of discrete-time signals and proceed to learn how to analyze data via the Fourier transform, how to manipulate data via digital filters and how to convert analog signals into digital. Z Transform and its applications also discussed in this course. Design and lab exercises are also significant components of the course

Course Objectives

- To introduce students basic techniques in designing and implementing digital signal processing systems.
- To learn basic method of pulse code modulation
- To explore the concept of convolution and correlation
- To learn basic methods of spectral analysis.
- To explore concept of CTFT,DTFT, DFT and FFT
- To teach students to design and implementation of various signal processing algorithms.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the fundamentals of digital signal processing
CO2	Understand various classes of digital signals
CO3	Illustrate the time domain representations of signals and systems
CO4	Explain the frequency analysis of signals

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓			✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓		✓	✓	✓

COURSE CONTENTS

Module 1: Signals and Signal Processing - Characterization and classification of Signals, Typical signal processing operations, Typical Signal Processing Applications, Advantage of Digital Signal Processing.

Module 2: Classification of signals –Introduction to vector space - The concept of frequency in continuous and discrete time signals -Sampling of analog signals – Sampling theorem – Quantization and Coding –Digital to analog conversion

Module 3: Time Domain Representation of signals and systems - Discrete time signals, Operations on sequences, Discrete time Systems, Linear Time invariant Discrete Time Systems-convolution sum – correlation of discrete time signals , Z-Transform

Module 4: Frequency Analysis of Signals - Frequency Analysis of Continuous Time Signals, Frequency Analysis of Discrete Time Signals, Fourier Transform of discrete time signals –Discrete Fourier Transform (DFT). FFT(Qualitative idea only)-Wavelet Transform - FIR and IIR Filters

Module X (For Additional Reading and Comprehension by the Students):

Digital Signal Processing Applications

- Audio signal processing
- Image and video processing
- Speech processing and recognition
- Telecommunications and digital communication systems
- Control systems and feedback control.

Advanced Topics in Digital Signal Processing

- Adaptive signal processing
- Multidimensional signal processing
- Wavelet transforms
- Compressed sensing and sparse signal processing
- Applications in biomedical signal processing

Multirate Signal Processing

- Decimation and interpolation
- Poly phase representation
- Filter banks and their applications
- Multistage implementation of decimators and interpolators

TEXT BOOKS

1.Proakis, John G. and Dimitris G. Manolakis. *Digital signal processing: principles algorithms and applications*. Pearson Education India, 2001.

.**REFERENCE BOOKS**

1.Roberts, Michael J. *Signals and systems: analysis using transform methods and MATLAB*. McGraw-Hill Higher Education, 2011.

2. Oppenheim, Alan V., and Ronald W. Schafer. Digital Signal Processing [by] Alan V. Oppenheim [and] Ronald W. Schafer.Prentice-Hall, 1975.

3. Antoniou, Andreas. *Digital signal processing*.McGraw-Hill, 2016.

4. Rabiner, Lawrence R., Bernard Gold, and C. K. Yuen. *Theory and application of digital signal processing*. Prentice-Hall, 2007

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes

1. What are the main differences between continuous-time signals and discrete-time signals? Provide examples of each.

2. Explain the concept of frequency in the context of both continuous-time and discrete-time signals. How is it different in these two domains?

3. What is the Nyquist-Shannon sampling theorem, and why is it important in the process of sampling analog signals to convert them into digital signals?

4. Describe the process of quantization and coding in digital signal processing. How does it impact the accuracy of the digitized signal?

5. What are the advantages of using digital signal processing techniques over analog signal processing methods? Provide specific examples of applications where digital signal processing excels.

6. Define the concept of convolution sum in discrete-time systems. How is it related to linear time-invariant (LTI) systems?

7. Explain the Z-transform and its significance in representing and analyzing discrete-time signals and systems.

8. What is the Fourier Transform, and how does it enable frequency analysis of continuous-time signals? Provide an example of its application in signal processing.

9. Discuss the Discrete Fourier Transform (DFT) and its practical importance in analyzing discrete-time signals in the frequency domain.

10. Compare and contrast Finite Impulse Response (FIR) filters and Infinite Impulse Response (IIR) filters in terms of their characteristics, applications, and advantages.

CORE COURSE

MSCSC02DSC09 ARTIFICIAL INTELIGENCE

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The objective of the course is to impart theoretical knowledge in the specialized area of Artificial Intelligence. The program focuses on building a comprehensive understanding on the fundamentals of Artificial Intelligence, ability to solve new problems, and a capacity to learn continually and interact with inter- disciplinary groups. This course has the potential to energies scientific and social advancement through technological innovation and entrepreneurship.

Course Objectives:

- To introduce basic principles that drive complex real-world intelligence applications.
- To introduce and discuss the basic concepts of AI Techniques
- To understand how the animal (especially human) intelligence is mimicked by the machines using AI algorithms.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Attain the knowledge about the importance of Artificial Intelligence.
CO2	Understand about AI techniques
CO3	Acquire the knowledge about Expert System
CO4	Familiarize the knowledge about Machine Learning

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓		✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction - Overview of AI applications. Introduction to representation and search.The Propositional calculus, Predicate Calculus, Using Inference Rules to produce Predicate Calculus expressions, Application.

Module 2:

Introduction to structure and Strategies for State Space search, Graph theory, Strategies for state spacesearch, using the State Space to Represent Reasoning with the Predicate calculus (Sate space description of a logical system, AND/OR Graph). Heuristic Search: introduction, Hill-Climbing and Dynamic Programming, The Best-first Search Algorithm, Admissibility, Monotonicity and Informedness.

Module 3:

Building Control Algorithm for State space search – Introduction, Production Systems, The blackboard architecture for Problem solving. Knowledge Representation – Issues, History of AI representational schemes, Conceptual Graphs, Alternatives to explicit Representation, Agent based and distributed problem solving. Strong Method Problem Solving –Overview of Expert System Technology, Rule Based Expert system, Model - Based, Case-Based and Hybrid Systems (Introduction to Model based reasoning, Introduction to Case Based Reasoning, Hybrid design), Introduction to Planning. Reasoning in Uncertain Situation (introduction), logic based Adductive Inference. Introduction to PROLOG.

Module 4:

Machine Learning: Symbol Based – Introduction, Frame –work. The ID3 Decision tree Induction algorithm. Inductive bias and Learnability, Knowledge and Learning, Unsupervised learning, Reinforcement Learning, Machine Learning: Connectionist – Introduction, foundations, Perceptron learning. Machine learning: Social and emergent: Models, The Genetic Algorithm, Artificial Life and Social based Learning.

Core Compulsory Readings

1. S. Russel and p. Norvig, Artificial intelligence – A Modern Approach, 3rdEdn, Pearson
2. D W Patterson, introduction to Artificial Intelligence and Expert Systems, PHI, 1990

Core Suggested Readings

1. George F Luger, Artificial Intelligence – Structures and Strategies for Complex probel solving, 5thEdn, pearson.
2. George J Klir and Bo yuan, Fuzzy sets and fuzzy logic: Theory and Applications, Pentice Hall India 1995

3. E. Rich, K. Knight, S B Nair, Artificial intelligence, 3rdEdn, McGraw Hill.

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. When you think of AI, what sort of challenges come to your mind?

2. Criticize Turing’s criteria for computer software being “intelligent”.

3. What are the major issues in knowledge representation?

4. Assume that you are developing an expert system for troubleshooting a complex electronic system. What is the source of uncertainty that can arise in this application?

5. Explain in detail about reinforcement learning with example.

6. In A* Algorithm, what do you mean by Estimated cost, h(n)?

CORE COURSE

MSCSC02DSC10 THEORY OF COMPUTATION

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course deals with the concepts of formal languages, grammar and automata. Topics covered include formal languages like regular language, context free language, context sensitive language and recursive enumerable language and their associated grammars. Students will learn how to construct abstract machines like finite automata, pushdown automata and standard Turing machine.

Course Objectives:

- Give basic idea on theory of computation and its applications.
- Impart knowledge on Finite Automata, regular languages, regular expression and regular grammar.
- Provide understanding on context free languages and pushdown automata
- Generate basic understanding on standard Turing machine and its variations.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Attain knowledge about Finite Automata.
CO2	Understand regular expressions, regular language and regular grammar
CO3	Obtain knowledge on context free grammar and pushdown automata
CO4	Attain knowledge about standard Turning Machine and hierarchy of formal languages

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to the Theory of computation and Finite Automata: Three basic concepts: languages, grammar & automata. Some applications. Finite automata: Deterministic Finite Acceptors, Nondeterministic Finite Acceptors, Equivalence of deterministic and nondeterministic finite acceptors, Reduction of the number of states in finite automata – State Equivalence method, table filling algorithm

Module 2: Regular Languages and Regular grammars: Regular expressions, connection between regular expressions and regular languages- state elimination method, Arden's lemma, regular grammars. Properties of Regular Languages: closure properties of regular languages, identifying non regular Language

Module 3: Context-free grammars & languages: Context-free grammars, parsing and ambiguity. Simplification of Context free Grammars and Normal forms : methods of transforming grammars, two important normal forms – CNF & GNF

Pushdown automata for context-free languages Non deterministic pushdown automata, PDA and context-free languages, deterministic pushdown automata and deterministic context-free languages. Properties of Context-Free Languages: pumping lemmas for context free languages, closure properties for context-free languages.

Module 4: Turing machine: Standard Turing machine, combining Turing machines for complicated tasks, Turing's thesis. Other models of Turing machine : Minor variations on the Turing machine theme, Turing machine with

complex storage, nondeterministic Turing machine, a universal Turing machine, Linear bounded automata.

A Hierarchy of Formal Languages and Automata: Recursive and Recursively Enumerable Languages, Unrestricted Grammars, Context-Sensitive Grammars and Languages, The Chomsky Hierarchy

Module X (For Additional Reading and Comprehension by the Students):

Mathematical preliminaries and notation, Proof techniques

Pumping lemma for linear languages

Limits of Algorithmic computation: Problems that cannot be solved by Turing machines, Undecidable Problems for Recursively enumerable Languages, The Post Correspondence problem

Turing Machine Models and Complexity: Language Families and Complexity Classes, The Complexity Classes P and NP

Finite State Transducer – Mealy Machines and Moore Machines

Core Compulsory Readings

1. An introduction to Formal Languages and Automata, Peter Linz, 4thedn, Narosa publishing House

Core Suggested Readings

1. John C Martin, Introduction to Languages and the Theory of Automata, McGraw Hill 1997
2. Mishra &Chandrasekharan, Theory of Computer Science : Automata, Languages andComputation, 3rd edn, PHI
3. Hopcroft, Motwani and Ullman, Introduction to automata theory, Languages and Computation, 3rdEdn,., Pearson

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What are formal languages? Give example.

2. Construct a minimal FA that accepts all the strings of a’s and b’s where every string ends with ab. ∑={a,b}

3. Explain the process of NFA to DFA conversion with the help of an example.

4. Construct a regular expression that generate all the strings of a’s and b’s where number of a’s in the string is even.

5. Construct a grammar that generates all the strings of a’s and b’s where every string contain at most 2a’s.

6. What is Chomsky Normal Form? Give example.

7. Explain the architecture of PDA.

8. Define Turing Machine. Explain its architecture.

CORE COURSE

MSCSC02DSC11 COMPUTER NETWORK AND LINUX ADMINISTRATION

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course provides an insight into the fundamental topics of Computer Networks and Linux Administration. A discussion on inter process communication and programming is provided. Configuration of Linux for network related activities are also discussed.

Course Objectives:

- To understand the basics of Computer Networks
- To acquire knowledge about the fundamentals of using Linux Operating System in a network environment
- To illustrate various, Inter Process Communication mechanisms
- To develop programs using various Inter Process Communication primitives
- To learn network configuration basics in Linux

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Explain the basics of Computer Networks and Linux Network Administration
CO2	Explain the basic steps and commands in administering Linux
CO3	Develop programs using various Inter Process Communication primitives
CO4	Explain network configuration basics in Linux

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓		✓		✓	✓
CO2	✓	✓			✓	✓
CO3		✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction, Basic concepts - Line configuration, Topology, Transmission mode, Categories of networks, Internetworks, Transmission media - Twisted pair Cable, Coaxial Cable, Optical Fiber. OSI and TCP/IP models, Functions of Layers in OSI and TCP/IP models. Network Devices: Hub, Switch, Router, Bridge, Gateway, Modem, Repeater, Access Point.

Module 2: Introduction to Linux: Linux and Unix, Common Linux Features, Advantage of Linux, Overview of Linux architecture, Linux files system, Linux standard directories. Commands for files and directories cd, ls, cp, rm, mkdir, rmdir, pwd, file, more, less, grep. Creating and viewing files using cat. Common administrative tasks: Obtaining Supervisor privileges, Setting file and directory permissions, Managing links, Users: concept, /etc/passwd file, /etc/shadow file, users groups and umask, adding, deleting and modifying user accounts. Major services in a UNIX system: init, login from terminals, syslog. LILO boot process and GRUB boot process.

Module 3: Inter Process Communication programming: Create a process - fork() system call, Parent and Child Process, Process ID, User and Group ID Half Duplex Unix Pipes, Named Pipes (First In First Out), Streams and messages, System V IPC: Message Queues, Shared memory. Sample programs for IPC that uses Pipes, Message Queues, and Shared memory. Socket Programming: Overview, TCP and UDP Sockets, Socket Address, Elementary Socket System Calls: socket, socket pair, bind, connect, listen, accept, send, sendto, recv, recvfrom, close, Byte ordering routines, Byte Operations, Address conversion routines, Simple client Programs that uses some reserved ports, Simple Client / Server Program using unreserved ports.

Module 4: TCP / IP Network Configuration: Introduction to TCP / IP network, Protocols, IP address, Hostname, Configuring a Host: setting the host name, assigning IP address, broad cast, net mask and name server address, Editing Host and network files, Interface Configuration: loopback interface, Ethernet interface, The SLIP and PPP interface, Configuring Gateway, Routing through gateway, Network commands: ifconfig, netstat, route. Dynamic IP Configuration: DHCP, Need for DHCP, Functions of DHCP. Monitor Network Connections: ping, traceroute, netstat with options i, r, l, p. Network Applications: File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP), Network File Systems (NFS), Network Information System(NIS), Hypertext Transfer Protocol (HTTP), Web Server.

Module X (For Additional Reading and Comprehension by the Students):

Transmission media: Satellite, Communication, Cellular Telephony, Terrestrial Microwave. History of Computer Networking and Internet. File system configuration: i-nodes, superblocks, ext3, reiserFS, mounting and unmounting local disks, /etc/fstab file. Core system services: init daemon, /etc/inittab file, telinit command, xinetd and inetd, enabling and disabling services. DHCP Server and Client Configuration. Email Protocols: Simple Mail Transfer Protocol (SMTP), Post office Protocol (POP), Multipurpose Internet Mail Extension (MIME). Domain Name Services (DNS): Working of DNS, Host name Resolution Name lookup with DNS, Reverse Lookup, Domain Name Servers and Zones, DNS database: SOA, NS, MX, A and PTR records, Secondary and primary DNS.

Core Compulsory Readings

1. Behrouz A. Forouzan, Data Communications and networking, Fourth Edition, McGraw Hill 2017
2. James F. Kurose and Keith W. Rose, Computer Networking A Top-Down Approach Featuring the Internet, Third Edition, Pearson Education
3. Wale Soyinka, Linux Administration A Beginner’s Guide, Fifth Edition, TMH
4. Linux Administrator STREET SMARTS A Real World Guide to Linux Certification Skills

Core Suggested Readings

1. Andrew S. Tanenbaum, Computer Networks, Fifth Edition, Prentice-Hall 2011
2. William Stallings, Data and Computer Communication, Tenth Edition, Prentice-Hall 2014
3. Evi Nemeth , et al, Linux Administration Hand Book , PHI 2018
4. Nicholas Wells, Linux Installation and Administration, Thomson Vikas 2003
5. Olaf Kirch& Terry Dawson, Linux Network Administrators Guide, O’relly, 2003
6. W Richard Stevens, Unix Network Programming, PHI, 2002

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration, Digital Learning, Team Work

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes

1. Write in brief about any one network device
2. Differentiate: FTP and TFTP
3. Explain any two methods for monitoring network connectivity
4. Illustrate IPC using Internet Domain Sockets

S2 - Ability Enhancement Courses (AEC)

Ability Enhancement Course

MCCSA02AEC01 ALGORITHM ANALYSIS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The objective of course is to impart theoretical knowledge in the specialized area of algorithm analysis. Study of algorithms is very substantial in classification of problems and their solutions based on complexity. Analysis of algorithms provides a means for choosing an appropriate algorithm for solving a problem at hand. The course provides an insight into all aspects of computational complexity and the use, design, analysis and experimentation of efficient algorithms. The better understanding paves way for successful implementations in various scientific applications.

Course Objectives:

- To write the performance of algorithms in mathematical terms
- To acquire knowledge for analyzing a given algorithm.

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Deep understanding about Time and Space Complexity
CO2	Acquire knowledge about the structure of the algorithm.
CO3	Obtain knowledge to analyze algorithm control structures and solving recurrence.
CO4	Attain information about Complexity Classes

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓		✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Important Problem Types: Sorting, Searching, String processing, Graph problems, Combinatorial problems, Geometric problems and Numerical problems. Basic Technique for Design of Efficient Algorithm: Brute Force approach (String matching), Divide-and-Conquer approach (Merge sort), Branch-and-Bound technique (Knapsack problem). Greedy approach (Kruskal's algorithm and Prim's Algorithm), Dynamic Programming (Longest Common Subsequence), Backtracking(Sum of subsets problem).

Module 2:

Importance of algorithm analysis, Time and Space Complexity. Growth of Functions: Asymptotic notations, Cost estimation based on key operations- Big Oh, Big Omega, Little Oh, Little Omega and Theta notations, Big Oh Ratio Theorem, Big Theta Ratio Theorem, Big Omega Ratio Theorem.

Module 3:

Analyzing Algorithm Control Structures, Solving Recurrences: Iteration Method, Substitution Method, The Recursion Tree Method, Master's Theorem, Problem solving using Master's Theorem Case 1, Case 2 and Case 3. Analysis of Strasser's algorithm for matrix multiplication, Analysis of Merge sort.

Module 4:

Complexity- Complexity Classes: P, NP, NP Hard and NP Complete problems. NP Completeness reductions for Travelling Salesman Problem and Hamiltonian Cycle. P versus NP problem.

Core Compulsory Readings

1. Thomas H Cormen, Charles E Leiserson, and Ronald L Rivest, Introduction to Algorithms, 3rd Edition, Prentice Hall of India Private Limited, New Delhi
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley
3. Pallaw, V K, Design and Analysis of Algorithms, Asian Books Private Ltd, 2012.

Core Suggested Readings

1.Pandey H M, Design and Analysis of Algorithms, University Science Press, 2013
2. Upadhyay, N, Design and Analysis of Algorithms, Sk Kataria & Sons, 2008.
3. U. Manber, Introduction to Algorithms: A Creative Approach, Addison Wesley
4. Gilles Brassard and Paul Bratley, Fundamentals of Algorithmics, Prentice-Hall of India
5. Goodman S E and Hedetniemi, Introduction to the Design and Analysis of Algorithms, Mcgraw Hill
6. Horowitz E and Sahni S, Fundamentals of Computer Algorithms, Galgotia Publications Pvt. Ltd
7. Oded Goldreich, P,NP and NP- Completeness, Cambridge University Press, 2011.
8. Donald Knuth, The Art of Computer Programming, Fundamental Algorithms,Volume- 1, Addison Wesley, 1997.
9. Sanjeev Arora and Boaz Borak, Computational Complexity- A Modern Approach, Cambridge University Press; 2009.

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	Percentage
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Ability Enhancement Course

MCCSA02AEC02 ARTIFICIAL INTELLIGENCE AND DAILY LIFE

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The objective of the course is to communicate the fundamental knowledge in the specialized area of Artificial Intelligence. The program focuses on building a comprehensive understanding on the basics of Artificial Intelligence and interact with inter- disciplinary groups.

Course Objectives:

- To understand the history of artificial intelligence(AI) and its foundations.
- To understand the basic principles of AI for solving daily life problems

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire knowledge aboutArtificial Intelligence
CO2	Obtain knowledge about Search and Control Strategies
CO3	Attain information about the fundamentals of Machine Learning
CO4	Demonstrate awareness and a fundamental understanding of various applications of AI techniques in intelligent agents, expert systems, Natural language Processing- machine learning models.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓		✓	✓	✓	✓
CO2		✓	✓	✓	✓	✓
CO3	✓	✓		✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction to Artificial Intelligence- History of AI- Advantages andDisadvantages of AI- Applications- AI domains.AI in real life, Expert system - Expert system development- Modern expert systems.

Module 2:

Search and Control Strategies- State- Space representation- Problem Solving - Heuristic Techniques – Hill Climbing – Simulated Annealing.

Module 3:

Machine Learning- Supervised and Unsupervised Algorithms- Neural Networks- Classification and Predictions model – Applications.

Module 4:

Natural Language Processing - Natural Language Processing Tasks -NLP Applications- Recommender System - Sentimental Analysis

Core Compulsory Readings

1. Stuart Russell and Peter Norvig, Artifcial Intelligence: A Modern Approach Third Edition Pearson Education 2010 Inc. ISBN: 978-0-13-604259-5.

2. D W Patterson, introduction to Artificial Intelligence and Expert Systems, PHI, 1990

Core Suggested Readings

1. E. Rich, K. Knight, S B Nair, Artificial intelligence, 3rdEdn, McGraw Hill.

2.https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_overview.htm

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

SEMESTER II

Ability Enhancement Course

MCCSA02AEC03 PYTHON PROGRAMMING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course mainly focuses to introduce the fundamental programming concept to the students from other disciplinaries. This course mainly focuses on the basic concept of programming constructs such as language elements, data types, operators control structures and data structure that will be useful for the programmers to learn the basic programming concept. Structure of this course is well organized way that it introduces the programing basic concept to advanced concepts such as modules, packages, GUI, and basics of NLP concepts for pattern matching and searching. After completing this course, the students acquire the ability to develop real life applications commonly useful for the society in may walk of life.

Course Objectives:

- Aims to impart basic programming skills to the learners in a simplest way.
- Impart knowledge on fundamental and advanced data structure concepts.
- Acquire the knowledge to impart various control structures to implement programming logic.
- Aware about the development of common GUI based applications in simple steps.
- Acquire the ability to analysis of data using NumPy and Pandas

SL #	Course Outcomes
CO1	Familiarize the different parts of Python programming and its applications in real world problems
CO2	Understanding the concept of different control structures and functions in Python.
CO3	Aware about the various data types and it operations and supporting methods
CO4	Make aware about the advanced concepts in Python such as reading and writing of data files , NumPy, Pandas and GUIs.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓	✓		✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓		✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction: History of Python Programming, Thrust Areas Of Python, Installing Anaconda Python Distribution, PyCharm IDE and Jupyter Notebook, Creating And Running First Python Project, **Parts of Python Programming Language**-identifiers, keywords, statements and expressions, variables, operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversion, The *typedef()* function and Is operator,

Module 2:

Control Flow Statement- Decision control flow statement *(if, if …else, if…elif…*, nested *if*), Loop (*while, for*),*continue*, *break* statements, Catching Exception Using *try* and *except* Statement

Functions- Built-In Functions Commonly used Modules, Function definition and calling the function, The *return* statement and *void* function, scope and life time of variables

Module 3:

Lists- Creating List, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions used on lists, list Methods. **Dictionaries**- Creating Dictionary, Accessing and Modifying *key:value* Pairs in Dictionaries, Built-In Functions used on Dictionaries, Dictionary Methods. **Tuples and Sets-** Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions used on Tuples, Relation between Tuples and Lists, Sets and Set Methods, Frozenset. **Strings-** Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Module 4:

Files- Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary files, Reading and Writing CSV Files, Introduction to NumPy and Pandas.

GUIs in Python: Root Window-Fonts and colors- Working with containers and canvas, Frames, Widgets, Button widgets, Arranging widgets in the Frame, Label Widget, Message Widget, Text widget, Scrollbar widgets, Check button widget, Radio button widget, Entry Widget, Spin box Widget, List Box Widget, Menu Widget.

Core Compulsory Readings

1. Gowrishankar S, Veena A, “Introduction to Python Programming”, 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Alberto Fernandez Villan, Mastering OpenCV 4 with Python, Packt Publishing Ltd.
3. Dr. R Nageswara Rao, Core Python Programming, 2nd edition, Dreamtech Publisher, 2019

Core Suggested Readings

1. Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 1st Edition,O'Reilly Media, 2017. ISBN – 13: 978-1491962299.

2. Wesley J. Chun, Core Python Programming, Second Edition, Publisher: Prentice Hall Pub

TEACHING LEARNING STRATEGIES

Lecturing, case study/mini projects, Team Learning, presenting seminars on selected topics, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print.

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1.What are the fundamentals data types in Python?

2.What are the different control structures in Python?

3.Explain the function and syntax of for loop control structure in Python with example.

4.What are functions? Explain how it differ from modules?

5.Explain the differences between modules and packages.

6. What are the different string operations in python? Explain.

7. Differentiate between mutable and immutable objects in Python,

8. Explain the basic operations on List.

9. What are the different types errors in a program? Explain each one.

10. What is CSV file?

11. Explain how will you read CSV files in Python.

12.What are Pandas? Explain its usage.

13. What are GUIs?

14. Explain the different Widgets in Python.

15. What is Canvas? Explain how it differ from Frames.

Ability Enhancement Course

MCCSA02AEC04 Data Processing with Python

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course includes an overview of the various tools available for writing codes and running the same in Python, and gets students coding quickly. It also provides hands-on coding exercises using commonly used data structures and writing custom functions after preprocessing the dataset.

Course Objectives:

- Understand how to perform some basic tasks to start exploring and analyzing the imported data set.
- Learn how to perform computations on the data to calculate basic descriptive statistical information
- Learn how to process data efficiently.
- Understand the visualization and analytics of data.

SL #	Course Outcomes
CO1	To understand python fundamentals and data types used in python
CO2	To learn basic statistical measurements required for machine learning
CO3	To familiarize preprocessing of data before processing.
CO4	To experience visualization of 1D and 2D data

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓			✓	✓	✓

COURSE CONTENTS

Module 1:

Python Fundamentals: Basic Concepts, Naming Variables, Operators and operands, Expressions, importing libraries, Redirecting the output, Data types, Lists, Working with files, Working with Directories; Big Data vs Data Science

Module 2:

Introduction to probability: discrete and continuous variables, probability distribution, Bayes Rule, Variance, Standard Deviation, Types of probability distributions, Algorithms designed using probability,

Module 3:

Finding datasets, Jupiter notebooks and loading data, pandas vs numpy, Saving, Dropping Null Values, User Defined Functions, Cleaning Dataset, Graphs and Statistics, Histogram, Working with rows and columns.

Module 4:

Introduction to basic plots, pandas vs matplotlib, visualizing 1D distributions, visualizing 2D distributions, Higher dimension visualizations.

Core Compulsory Readings

1. Wes McKinney, “Python for Data Analysis”, 2nd Edition, O’Reilly
2. Joel Grus,” Data Science from Scratch” First Edition, April 2015

3. McKinney, W. (2012). Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. " O'Reilly Media, Inc.".

TEACHING LEARNING STRATEGIES

Lecturing, case study/mini projects, Team Learning, presenting seminars on selected topics, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print.

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions

1. What skills are necessary for a Data Scientist?
   a. Statistics
   b. Deep Learning
   c. Linear Algebra
   d. All of the above
2. CLI stands for ………..
   a. Command Line Interface
   b. Command Language Interface
   c. Command Line intercom
   d. None of the Above

3. Give one example where both false positives and false negatives are important equally?
4. Write the difference between data analytics and data science.
5. What are data science tools?
6. List different data types available in python with examples for each.

Ability Enhancement Course

MCCSA02AEC05 FUNDAMENTALS OF BIG DATA

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course mainly focuses to introduce the fundamentals of big data analytics concept using Python programming language. First part of this course familiarizes basic concepts of python programming language and subsequently introduces the big data analytics concepts and machine learning algorithms for developing mathematical/statistical models for analyzing big data using NumPy and Pandas in Python. After completing this course, the students acquire the basic ability to develop real life applications for extracting hidden patterns/trends in a big data environment that is commonly useful for the society in may walk of life.

Course Objectives:

- Aims to impart basic programming skills to the learners for developing mathematical models for big data analysis.
- Impart knowledge on fundamental and advanced data structure concepts in Python programming language suitable for big data analysis.
- Aim to impart to knowledge in bigdata platforms and developing mathematical/statistical models for big data analysis using machine learning concepts.
- Acquire the ability to analysis of data using NumPy and Pandas

SL #	Course Outcomes
CO1	Familiarize the different parts of Python programming and its applications in real world problems
CO2	Understanding the concept of different control structures and functions in Python.
CO3	Aware about the different types of data files and an overview of various data analytics methods.
CO4	Acquire the knowledge of different machine learning algorithms and its implementation for building efficient data analytics models for real life.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓	✓		✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

History of Python Programming, Thrust Areas Of Python, Installing Anaconda Python Distribution, PyCharm IDE and Jupyter Notebook, Creating And Running First Python Project, **Parts of Python Programming Language-**identifiers, keywords, statements and expressions, variables, operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversion, The *typedef()* function and Is operator,

Module 2:

Control Flow Statement- Decision control flow statement *(if, if ...else, if...elif…*, nested *if*), Loop (*while, for*),*continue, break* statements, Catching Exception Using *try* and *except* Statement

Functions- Built-In Functions,commonly used Modules, Function definition and calling the function, The *return* statement and *void* function, scope and life time of variables

Module 3:

Files: - Types of Files: -Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary files, Reading and Writing CSV Files, Introduction to NumPy and Pandas.

Data Analytics Overview - Statistical Computing - Mathematical Computing Using NumPy - Data Manipulation with Pandas - Data Visualization with Python - Introduction to Model Building for Evaluation

Module 4:

Supervised Learning - Classification, Naive Bayes, KNN, Linear Regression. Unsupervised

Learning - Clustering, Hierarchical algorithms – Agglomerative algorithm, Partitional algorithms - K- Means. Association Rule Mining - Apriori algorithm.

Core Compulsory Readings

1. Bart Baesens," Analytics in a Big Data World: The Essential Guide to Data Science and its Business Intelligence and Analytic Trends", John Wiley & Sons, 2013.

2. David Dietrich, "EMC Education Services, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", John Wiley & Sons, 2015.

3.Dr. R Nageswara Rao, Core Python Programming, 2nd edition, Dreamtech Publisher, 2019

TEACHING LEARNING STRATEGIES

Lecturing, case study/mini projects, Team Learning, presenting seminars on selected topics, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print.

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1.What are the fundamentals data types in Python?

2.What are the different control structures in Python?

3.Explain the function and syntax of for loop control structure in Python with example.

4.What are functions? Explain how it differ from modules?

5.Explain the differences between modules and packages.

6. What are the different string operations in python? Explain.

7. What is CSV file?

8. Explain how will you read CSV files in Python.

9.What are Pandas? Explain its usage.

10. What is NumPy?

11. Explain the use of pandas in Big data analytics.

12.What is Big Data?

13. What are the different tools in Python for data visualization? Explain.

14. What is KNN?

15. Explain about supervised and unsupervised learning with suitable examples.

Ability Enhancement Course

MCCSA02AEC06 Fundamentals of Programming Methodology

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course description: Computer Science is all about developing correct and efficient solutions for our day to day problems. The process of developing solutions is not centered on learning a programming language and doing coding straight away. Instead a blueprint of the proposed solution should be outlined and it should be tested for correctness. Once a proposed blueprint leads to a correct solution, it can be implemented using a suitable programming language. Objective of this course is to impart knowledge to the learner about building the blueprint of a solution. Learners are also exposed to implementing the solutions using the C programming language.

Course Objectives:

- To impart knowledge about various constructs for developing solutions
- To become familiar with using the various constructs to develop solutions
- To compare and contrast various constructs for solution development for selection
- To compare and contrast various constructs for solution development for iteration
- To implement solutions using C programming language

SL #	Course Outcomes
CO1	Identify the historical evolution of Computer Science in the perspective of problem solving
CO2	Illustrate the foundations of developing solutions using flowcharts and algorithms
CO3	Develop solutions using various selection constructs and implement them in C programming language
CO4	Develop solutions using various iteration constructs implement them in C programming language

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓	✓	✓		✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: History and evolution of Computer Science in the perspective of problem solving - Problem Solving – Fundamental Constructs for Building Solutions. Flowchart - System Chart - Program Charts – Variables, data names, programming statements – Steps for Developing Solutions - Preparing Solutions using Flowcharts: Conventions - Structure - Symbols. Preparing Solutions using Algorithms - Conventions - Top Down Design – Structure of Algorithms. Case Study: Developing flowcharts and algorithms for problems using sequence construct

Module 2: Program: Characteristics - Modular Approach - Style - Documentation and Maintenance - Compilers and Interpreters - Preparing, Running and Debugging Programs - Types of Errors. Fundamentals of C Language: Evolution and Features - Program Structure - Elements - Constructs. Character Set, Tokens, Keywords, Identifier. Data Types, Constants, Symbolic Constants, Variables, Data Input and Output, Statements - Assignment statements. Operators in C: arithmetic, relational, logical, assignment, auto increment, auto decrement, conditional, comma operators. Precedence of operators - expressions – evaluation of expressions, type conversion in expressions – precedence and associativity. Case Study: Preparing C programs for solutions developed in Unit I.

Module 3: Selection Constructs: Simple if - if else - if else if ladder - switch. Branching statements: break, goto. Case study: Developing solutions (flowcharts and algorithms) for problems using various selection constructs - Comparative Study of various Selection Constructs - Converting a solution using one selection construct with other selection constructs.

Module 4: Iteration Constructs: Top Tested Vs Bottom Tested - while - for - do while - Nesting of loops - skipping breaking loops. Arrays - 1D and 2D - strings – processing of strings - string manipulation functions. Case study: Developing solutions (flowcharts and algorithms) for problems using various iteration constructs - Comparative Study of various iteration constructs - Converting a solution using one iteration construct with other iteration constructs.

Core Compulsory Readings

1. J.B Dixit, Computer Fundamentals and Programming in C, Firewall Media

2. Anil Bikas Chaudhuri, The Art Of Programming Through Flowcharts Algorithms, Laxmi Publications, New Delhi.

3. Maureen Spraknle and Jim Hubbard, Problem Solving and Programming Concepts, Pearson

4. E Balagruswamy, Programming in ANSI C, TMH, 5th Edition

Core Supplementary Reading

1. R G Dromey, How to Solve by Computer, Pearson Education, 5th Edition

2. Brian W. Kernighan and Dennis M. Ritchie, C Programming Language, PHI

Kanetkar, Let Us C, BPB Publications, 8th Edition

TEACHING LEARNING STRATEGIES

Lecturing, case study/mini projects, Team Learning, presenting seminars on selected topics, Digital Learning

MODE OF TRANSACTION

Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print.

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

S2 -Skill Enhancement Courses (SEC)

Skill Enhancement Course

MCCSA02SEC01 Basics of Machine Learning

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Machine Learning is the study of how to build computer systems that learn from experience. It is a subfield of Artificial Intelligence and intersects with statistics, cognitive science, information theory, optimization, and probability theory. The course will explain how to build systems that learn and adapt using examples from real-world applications. The main topics include neural networks, decision trees, KNN, unsupervised learning, and reinforcement learning.

Course Objectives:

- To understand the principles and techniques in Machine Learning.
- Helps to understand statistical techniques and different algorithms in machine learning.
- Produce Python code to statistically analyse a dataset using classification and regression techniques.
- Compare and contrast evaluation metrics

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	To understand the basic concepts in machine as well as python programming language
CO2	To familiarize statistical methods and how to apply the same in machine learning
CO3	To learn different types of machine learning algorithms
CO4	To experience the implementation of machine learning algorithms and calculation of accuracy.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction to Machine Learning; Components of Learning; Categories of Machine Learning-Supervised Learning, Unsupervised Learning and Reinforcement Learning; Introduction to Python: NumPy, Matplotlib, Pandas

Module 2:

Statistical Techniques-Review on probability, Regression, Classification, Clustering, Probability Theories, Decision Trees; IDEs, Dimensionality Reduction; Main challenges of Machine Learning, Deep Learning

Module 3:

Decision Tree Classifiers, Classification and Regression trees; Regression-Linear Regression, Logistic Regression; Neural Networks-Introduction, Perception; Support Vector Machines; K Nearest Neighbors; Introduction to Clustering-K Means Clustering

Module 4:

Machine Learning Steps: Collection of data, Preparation, Model Selection, Training, Evaluation, Parameter Tuning, Making Predictions; Implementation; Confusion Matrix, Accuracy, Precision, Recall, ROC curve and AUC.

Core Compulsory Readings

1. Steven S. Skiena, “The Data Science Design Manual”, Springer 2017.

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions

1. Machine Learning is a subset of which of the following:
   a. Artificial Intelligence
   b. Data Learning
   c. Deep Learning
   d. None of the Above
2. Decision Tree cannot be used for clustering. Choose whether true or false
   a. True
   b. False
3. Which technique is used to predict categorical responses?
4. List any four classification algorithm.
5. How can outlier values be treated?
6. Define ensemble learning.
7. What are the assumptions required for a linear regression?
8. List the differences between supervised and unsupervised learning.
9. What is deep learning? What is the difference between deep learning and machine learning?
10. List some of the importance of dimensionality reduction.
11. Explain Neural Network fundamentals.

Skill Enhancement Course

MCCSA02SEC02 Data Science Fundamentals

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

A Data Science course covers basic and advanced concepts of data analytics, machine learning, statistics, and programming languages like Python or R. It also teaches students how to clean a dataset, and interpret large datasets and identify patterns to create predictive models.

Course Objectives:

- To provide strong foundation for data science and application area related to it and understand the underlying core concepts and emerging technologies in data science.
- Demonstrate an understanding of statistics and machine learning concepts that are vital for data science
- Explain how data is collected, managed and stored for data science
- To understand how to perform data analysis and visualization task.

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	To learn basic idea about data science fundamentals, machine learning and statistical probabilities.
CO2	To understand different types of data such as structured and unstructured data.
CO3	To familiarize different classification algorithms.
CO4	To experiment different algorithms using python and calculate the accuracy.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓	✓	✓	✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction to Data Science: Definition, Big data, populations and samples, exploratory data analysis, properties of data, Mathematical and Statistical Skills, Data Analysis Tools, Data Science Applications

Module 2:

Types of data, Application areas of Data Science, Data Science process, Machine Learning: Introduction, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Introduction to Statistics.

Module 3:

Basic Machine Learning Algorithms: Linear Regression, SVM, Naïve Bayes, example program implementation, Data acquisition, Transformation into standardized format, Data Cleaning, Data Reduction, Data Integration, Data Transformation, Normalization.

Module 4:

Algorithm design and analysis, Data, Database Table, Python, Confusion Matrix, Data Handling and Visualization, Different Chart types, Plotting Functions, Solving problem with Data Science.

Core Compulsory Readings

Text Book

1. Rachel Schutt & O'neil, "Doing Data Science", Straight Talk from The Frontline O'REILLY, ISBN:978-1-449-35865-5, 1st edition, October 2013.

2. Joel Grus,” Data Science from Scratch” First Edition, April 2015

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test outcomes:

1. Identify the language which is used in data science.
   a. R
   b. C++
   c. Java
   d. Ruby
2. Total groups in which data can be characterized is:
   a) 4
   b) 2
   c) 1
   d) 3
3. What are some of the techniques used for sampling?
4. Explain the major Components of Data Science.
5. What is Data Science? List the differences between supervised and unsupervised learning.
6. List some of the applications of data science in the real world scenario.
7. Write the difference between a box plot and a histogram.

Skill Enhancement Course

MCCSA02SEC03 OPTIMIZATION TECHNIQUES

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Optimization, also known as mathematical programming, collection of mathematical principles and methods used for solving quantitative problems in various disciplines, including physics, biology, engineering, economics, and business. The subject grew from a insight that quantitative problems in obviously different disciplines have important mathematical elements in common.

Course Objectives:

- To understand the need of optimization
- To study various optimization techniques

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the importance of optimization technique
CO2	Discuss Optimum design concepts
CO3	Solve the Linear Programming models using graphical and simplex methods
CO4	Evaluate different algorithmic methods for solving constrained and unconstrained optimization problems

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓			✓	✓
CO2	✓	✓	✓		✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Optimization: Introduction, Statement of an Optimization problem, formulation of Optimal Problem, Types of Optimization problem.

Module 2:

Optimum design concepts: Definition of Global and Local optima, Optimality criteria, Convexity and concavity of functions of one and two variables ,Lagrangian function, Hessian matrix formulation

Module 3:

Linear programming: Standard form of Linear Programming Problem, Canonical form, Elementary operations, Graphical method for two variable optimization problem, Simplex method, Karmarkar's projective scaling method.

Module 4:

Optimization algorithms for solving unconstrained optimization problems – Gradient based method: Cauchy's steepest descent method, Newton's method, Conjugate gradient method. Optimization algorithms for solving constrained optimization problems– direct methods – penalty function methods – steepest descent method.

Core Compulsory Readings

1. G. Hadley, Linear programming, Narosa Publishing House, New Delhi, ISBN

13: 9788185015910.

2. Singiresu S. Rao, Engineering Optimization: Theory and Practice by John Wiley and Sons,(5th edition),ISBN: 978-1-119-55479-3

Core Suggested Readings

1. Shikare MM, Waphare BN, Combinatorial Optimization, Narosa Publication (2004)

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Skill Enhancement Course

MCCSA02SEC04 Scientific Computing

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course on Scientific Computing is designed to provide students with a comprehensive introduction to computational techniques used in various scientific disciplines. Students will gain practical knowledge of programming languages commonly used in scientific computing, numerical methods, data manipulation, and high-performance computing. The course will emphasize hands-on experience through programming exercises and projects, enabling students to apply the acquired skills to solve real-world scientific problems.

Course Objectives:

- Introduce students to the fundamentals of scientific computing and its significance in diverse scientific domains.
- Familiarize students with programming languages and libraries commonly used for scientific computations, such as Python, Julia, or MATLAB.
- Equip students with essential numerical analysis techniques and optimization methods for solving scientific problems efficiently and accurately.
- Develop students' skills in data manipulation, analysis, and visualization to gain insights from scientific datasets.

Learning Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Demonstrate a solid understanding of scientific computing principles, methodologies, and applications.
CO2	Choose an appropriate programming language and effectively implement numerical methods to solve scientific problems.
CO3	Analyze and interpret scientific data using statistical techniques and visualization tools.
CO4	Apply numerical techniques to solve linear and non-linear equations, eigen value problems, and differential equations.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓		✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction to Scientific Computing. Overview of Scientific Computing: Role, significance, and applications in various scientific disciplines. Essential programming concepts: Variables, data types, loops, conditionals, functions, and basic I/O. Introduction to a programming language (Python recommended): Syntax, data structures, and libraries for scientific computing. Numerical methods: Root finding, interpolation, integration, and differentiation. Data visualization: Plotting techniques and tools for presenting scientific data effectively.

Module 2:

Data handling and manipulation using libraries such as NumPy, pandas, or equivalent. Statistical analysis of data, including hypothesis testing and regression. Advanced data visualization techniques for scientific presentations. Time-series analysis and Fourier transforms for signal processing. Data cleaning and preprocessing for scientific datasets.

Module 3:

Numerical Linear Algebra. Matrix and vector operations: Addition, subtraction, multiplication, and division. Solving linear systems: Gaussian elimination, LU decomposition, and iterative methods (Jacobi, Gauss-Seidel). Eigenvalue and eigenvector computation.

Module 4:

Differential Equations and Optimization. Ordinary Differential Equations (ODEs): First-order and higher-order ODEs, initial value problems, and boundary value problems. Numerical integration methods: Euler's method, Runge-Kutta methods (finite difference, finite element, etc.), and applications.

Core Compulsory Readings

1. Sastry S.S, INTRODUCTORY METHODS OF NUMERICAL ANALYSIS, 5TH EDN,Prentice Hall India Learning Private Limited, 2012, ISBN: 978-8120345928
2. Germund Dahlquist, Ake Bjorck, Numerical Methods in Scientific Computing, SIAM, 2008, ISBN:9780898716443

Core Suggested Readings

1. Bertil Gustafsson, Fundamentals of Scientific Computing,Springer Science & Business Media, 2011, ISBN: 9783642194948

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Skill Enhancement Course

MCCSA02SEC05 The Art of E - Documentation using Latex

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: This course is intended to impart the foundations of document preparation using Latex. Latex is a simple but flexible and powerful document editor that can be used to create documents of varying purposes. It can also be used to create slides for presentations.

Course Objectives:

- To impart knowledge about the structure of Latex documents
- To become familiar with using the various options to type set and format contents in Latex
- To create documents with tables, list and images in Latex
- To add citations and references for a scientific document

Learning Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Identify the anatomy of a Latex document and prepare simple documents in Latex
CO2	Illustrate the foundations of typesetting and formatting in Latex
CO3	Prepare documents with lists, tables and images and to create references and citations
CO4	Create slides using beamer

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓		✓	✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓		✓	✓	✓	✓
CO4	✓	✓		✓	✓	✓

COURSE CONTENTS

Module 1: WYSIWYG Editors Vs Latex. Latex: History - Advantages - Editors (Windows / Mac / Ubuntu - Online). Anatomy of Latex Document - Common Document Classes - Paper sizes - Commonly Used Packages - Setting margins. Document Creation - Beginning Document - Sections - Adding Text - Inserting blank lines - Ending Document

Module 2:. Typesetting Document - Fonts, symbols, indenting, paragraphs, line spacing, word spacing, titles and subtitles, adding colors to text and entire page, adding bullets and numbered items. Adding header and footer, changing the page orientation, dividing the document into multiple columns

Module 3: Creating Tables: Setting Columns - Merging Rows and Columns - Various Styles and Orientation. Adding Images - Graphics Packages - Rotation - Scaling . Drawing Matrices - Adding Mathematical Symbols and Equations. Adding Bibliography, Cross References and Citations - Adding Table of Contents, Figures and Tables

Module 4: Beamer - Creating Slides - Adding frames - Dividing the slide into multiple columns - Adding different blocks - Adding Tables and Graphics. Case Study: Preparation of Resume, Official Letters, Book, Article, Homework assignment

Core Compulsory Readings

1. Free online introduction to LaTeX Available at https://www.overleaf.com/learn/latex/Free_online_introduction_to_LaTeX_(part_1)
2. Learning Latex, D.F.Griffits, D.J.Higham, Siam, Philadelpha, 1997
3. Latex, https://en.wikibooks.org/wiki/LaTeX

Core Suggested Readings

1. Online materials related to document preparation using Latex and slide preparation using Beamer

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Skill Enhancement Course

MCCSA02SEC06 Fundamentals of Digital Skilling using Google Workspace for Education

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: This course is intended to provide a fundamental understanding about the services offered by the Google Workspace for Education. Learners will get an exposure to how to use the services such as Google Drive, Google Docs, Google Sheet and Google Slides for their personal and collaborative learning strategies.

Course Objectives:

- To familiarize with about the Google Workspace for Education
- To understand the fundamentals of Gmail
- To get familiar with Google Drive
- To create documents using Google Doc and Google Sheet
- To create slides using Google Slides

Learning Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Identify the services offered by Google Workspace for Education and get exposed to the fundamentals of Gmail
CO2	Illustrate the foundations of preparing documents using Google Docs
CO3	Prepare documents using Google Sheets
CO4	Prepare slides using Google Slides

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓		✓	✓	✓	✓
CO2	✓		✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓		✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Google Workplace for Education: Gmail, Google Calendar, Google Drive, Google Docs, Google Sheets, Google Slides. Gmail - Creation of ID - Composing Messages - Folders - Labels - Simple Customizations.

Module 2:. Google Drive: Uses - Accessing - Viewing Drive Contents - Creating Folders - Creating Documents - Making Copies - Uploading Documents - Downloading Folder Contents - Sharing Folders - Erasing Documents - Creating and Managing Workspaces - Searching Contents - Managing Storage.

Module 3: Google Doc: Creation - Saving - Typesetting and Formatting - Images - Tables - Charts - Line - Emoji - Smart Chips - Dropdown - Footnote - Header and Footer - Page Numbers - Indentation - Lists - Checklists - Creating Links - Watermark - Page Orientation - Spelling and Grammar Check - Citations - Modes (View / Review / Edit) - Voice Typing - Print - Download - Sharing.

Module 3: Google Sheet: Creation - Components - Cells and Addressing - Navigating. Entering and Editing Data: Text - Equations - Typesetting and Formatting. Saving Worksheet. Editing Worksheet Data - Moving and copying data - Moving and copying equations - Inserting and deleting ranges, rows, and columns. Formatting: Text - Numbers - Row and Column -- Conditional Format - Borders and Colours. Worksheets: Naming, Copying, Creating, Removing Charts: Types - Pie Chart - Bar Chart. Data - Sorting - Filters - Validation - Removal of Duplicates. - Page Orientation - Spelling and Grammar Check - Print - Download - Sharing.

Module 4: Google Slide: - Creating Slides - Adding Image, Text Box, Audio, Vedio, Shape, Chart, Diagram. Typesetting and Formatting - Lists - Checkboxes. Slides: Ordering - Changing Background - Deleting - Transition. Share - Print - Download

Core Compulsory Readings

1. Learn the basics, https://workspace.google.com/intl/en_in/training/

Core Suggested Readings

1. Online materials related to Google Workplace for Education

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Skill Enhancement Course

MCCSA02SEC07 Image Processing using Python

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This comprehensive four-module course on " Image Processing using Python " is designed to provide students with a deep understanding and practical skills in image manipulation, enhancement, and analysis. Through hands-on exercises and real-world applications, students will navigate the powerful landscape of Python libraries, including PIL and OpenCV, to unlock the potential of images for various domains.

Course Objectives:

- Introduce students to Python libraries for image processing, emphasizing PIL and OpenCV.
- Provide hands-on experience in basic image manipulation and enhancement techniques.
- Master basic and advanced image filtering techniques for enhanced image quality.
- Build an image classification model using TensorFlow for categorizing images.
- Understand image transformation techniques, focusing on Fourier Transform.
- Provide an introduction to deep learning for image processing.

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	proficiency in using Python libraries for image processing
CO2	Comprehensive understanding of image histograms, image smoothing, sharpening techniques, and various filtering methods
CO3	Proficiency in thresholding and region-based segmentation techniques.
CO4	Ability to complete a real-world image processing project, highlighting students' ability to solve practical problems in diverse scenario

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	
CO3	✓	✓		✓	✓	
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction to Python libraries for image processing, Basic image manipulation and enhancement techniques. Introduction to Python for Image Processing. Image representation and basic operations using Python libraries (e.g., PIL or OpenCV). Reading and displaying images in Python. Basic image processing operations: resizing, cropping, and rotating.

Module 2:

Understanding image histograms, applying image smoothing and sharpening techniques, understanding and applying basic and advanced image filtering techniques. Image restoration techniques, Edge detection methods. Feature extraction and representation. Practical applications of segmentation and feature extraction. Object detection and recognition.

Module 3:

Thresholding and region-based segmentation techniques. Image classification model with TensorFlow. Preprocessing, Segmentation and Registration of medical images. Understanding 3D image processing, image visualization and manipulation.

Module 4:

Image compression technique, JPEG and Wavelet-based compression technique. Introduction to image steganography, hiding data and extracting hidden data from images using Python. Image transformation techniques (e.g., Fourier Transform). Introduction to deep learning for image processing.

Core Compulsory Readings

1. "Python Imaging Library (PIL) Handbook" by Fredrik Lundh
2. "Digital Image Processing" by Rafael C. Gonzalez and Richard E. Woods
3. "OpenCV with Python By Example" by Prateek Joshi
4. "Hands-On Image Processing with Python" by Sandipan Dey

Core Suggested Readings

1. "Image Processing in Python with OpenCV" by Gabriel Garrido Calvo

2. "Python Machine Learning" by Sebastian Raschka and Vahid Mirjalili

3. "Image Processing and Acquisition using Python" by Ravishankar Chityala, Sridevi Pudipeddi, and G.V. Sridhar

4. "Introduction to Deep Learning" by Andreas C. Müller and Sarah Guido

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes:

1. Define an image histogram and explain how it is useful in image processing. Provide a simple example.
2. Explain the steps involved in building a basic image classification model using TensorFlow.
3. Explain the basic concept of Fourier Transform in image transformation. How does it alter the representation of an image?
4. What is the fundamental idea behind deep learning in image processing?
5. Provide a basic example of how deep learning can enhance traditional image processing methods.

Skill Enhancement Course

MCCSA02SEC08 Fundamentals of Electrical and Electronics Engineering

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The "Fundamentals of Electrical and Electronics" course is a comprehensive skill development program designed to introduce participants to the core principles of electrical and electronic engineering. The course aims to provide a solid foundation in electrical concepts, electronic devices, electrical machines, and power systems. Through a combination of theoretical knowledge and hands-on practical exercises, participants will gain the necessary skills to analyse, design, and troubleshoot basic electrical and electronic circuits. The course will also cover safety measures and regulations to ensure a safe working environment.

Course Objectives:

- Understand Basic Electrical Concepts: The course aims to familiarize participants with the fundamental principles of electricity, including voltage, current, resistance, and power, enabling them to analyse simple electrical circuits.
- Explore Electronic Devices and Applications: Participants will learn about semiconductor devices such as diodes and transistors and their applications in rectification, amplification, and digital logic circuits.
- Acquire Knowledge of Electrical Machines and Power Systems: The course will introduce different types of electrical machines, transformers, and power transmission systems, enabling participants to comprehend their working principles and applications.
- Develop Circuit Design and Troubleshooting Skills: Participants will be exposed to circuit design fundamentals, schematic creation, and PCB layout using software tools. Additionally, they will learn troubleshooting techniques using multimeters and oscilloscopes to identify and fix common circuit issues.

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Upon completion of this course, participants will be able to explain the basic concepts of electricity, including voltage, current, resistance, and power, and apply Ohm's Law to calculate these values in simple circuits.
CO2	Participants will be able to identify and describe various electronic components and understand their applications in practical circuits, such as rectifiers, amplifiers, and digital logic circuits.
CO3	After completing the course, participants will be able to recognize different types of electrical machines, including DC motors, AC induction motors, and transformers, and comprehend their roles in various industrial and domestic applications.
CO4	By the end of the course, participants will have acquired the skills to design basic electrical and electronic circuits, create schematics and PCB layouts using software tools, and troubleshoot common circuit issues using multimeters

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1				✓	✓	✓
CO2	✓		✓	✓	✓	✓
CO3				✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Electrical Concepts

Basics of Electricity: Introduction to electrons, protons, and atomic structure. Understanding electric charge, current, and voltage. Differentiating between AC and DC currents. 1.2 Circuit Components and Symbols: Identify and describe passive components (resistors, capacitors, inductors). Recognize and explain active components (diodes, transistors). Introduction to circuit symbols and diagrams. Ohm's Law and Power: Understanding Ohm's Law and its applications. Calculating voltage, current, and resistance in circuits. Power calculations and power rating of components. 1.4 Circuit Analysis Techniques: Series and parallel circuits analysis. Kirchhoff's laws and their application. Thevenin and Norton equivalents

Module 2: Electronic Devices and Applications. Semiconductors and Diodes: Introduction to semiconductors and their properties. Understanding diodes and their applications (rectifiers, zener diodes). 2.2 Transistors and Amplifiers: Types of transistors (BJT and MOSFET) and their characteristics. Transistor as an amplifier: common emitter, common collector, common base configurations. Digital Electronics: Basics of digital circuits and logic gates. Electronic Sensors and Actuators: Understanding different types of sensors (temperature, light, proximity) Introduction to actuators (motors, relays) and their applications

Module 3: Electrical Machines and Power Systems

3.1 Introduction to Electrical Machines: Overview of different types of electrical machines (DC motors, AC motors, transformers). Working principles and applications. Transformers and Power Transmission: Understanding transformers and their types Introduction to power transmission and distribution systems. 3.3 Electric Motors: Basics of DC motors and their characteristics. Introduction to AC induction motors and synchronous motors

Module 4: Circuit Design and Troubleshooting. Circuit Design Fundamentals: Overview of circuit design process and best practices. Schematic and PCB design introduction using software tools. Troubleshooting Techniques: Identifying and fixing common circuit issues.

Core Compulsory Readings

1. "Electric Circuits" Author: James W. Nilsson, Susan A. Riedel Publisher: Pearson Year: 2020
2. "Electronic Devices and Circuit Theory" Author: Robert L. Boylestad, Louis Nashelsky Publisher: Pearson Year: 2019
3. "Electric Machinery Fundamentals" Author: Stephen J. Chapman Publisher: McGraw-Hill Education Year: 2021

Core Suggested Readings

1. "Electrical Engineering: Principles & Applications" Author: Allan R. Hambley Publisher: Pearson Year: 2019

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Skill Enhancement Course

MCCSA02SEC09 Data Analysis and Plotting

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course provides a comprehensive introduction to data analysis and plotting using Origin Software. Participants will learn fundamental data manipulation techniques, exploratory data analysis (EDA), and advanced data analysis methods. Through hands-on exercises and projects, students will gain proficiency in creating informative and visually appealing graphs, making data-driven decisions, and presenting their findings effectively.

Course Objectives:

- To equip participants with essential data analysis skills using Origin Software.
- To enhance participants' ability to interpret and visualize complex datasets accurately.
- To enable participants to apply statistical techniques for data-driven decision-making.
- To empower participants to create dynamic and interactive graphs for effective data communication.

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Identify and apply data analysis techniques: Participants will demonstrate an understanding of various data analysis techniques, including data manipulation, descriptive statistics, filtering, and basic plotting. They will proficiently apply these techniques using Origin Software to gain insights from raw data.
CO2	Analyze and interpret data using Origin Software: Participants will be able to perform exploratory data analysis (EDA) using Origin Software. They will learn to create and interpret histograms, box plots, scatter plots, and other visualization tools to assess data distributions, relationships, and patterns.
CO3	Apply advanced data analysis methods: Participants will learn and implement advanced data analysis methods, such as statistical hypothesis testing, curve fitting, and peak analysis, using Origin Software. They will understand how to make data-driven decisions based on these analyses.
CO4	Design and create compelling data visualizations: Participants will acquire skills to design publication-quality graphs and visualizations using Origin Software. They will learn to customize graph styles, colors, annotations, and multimedia elements to create engaging and impactful presentations of their data analysis results.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓			✓	✓	✓
CO3		✓		✓	✓	✓
CO4		✓		✓	✓	✓

COURSE CONTENTS

Module 1:

Introduction to Data Analysis and plotting Software Understanding the importance of data analysis in various fields. Overview of plotting Software and its capabilities. Installing and setting up plotting Software. Importing data into software from different sources. Data manipulation and cleaning techniques in plotting Software. Introduction to basic plotting and visualization options in plotting Software.

Module 2:

Exploratory Data Analysis (EDA) with Origin. Understanding the concept of EDA and its role in data analysis. Utilizing descriptive statistics to summarize data. Visualizing data distribution using histograms, box plots, and density plots. Exploring relationships between variables using scatter plots and correlation analysis. Implementing data filtering and conditional formatting in plotting software. Customizing plot styles, colors, and annotations for better visualization.

Module 3:

Advanced Data Analysis. Performing statistical analyses in Origin, including t-tests, ANOVA, and regression. Using built-in analysis tools like peak analysis, curve fitting, and smoothing. Handling missing data and dealing with outliers in Origin. Applying data transformations and normalization techniques. Spectral analysis of data.

Module 4:

Advanced Plotting and Data Visualization. Understanding different plot types: 2D and 3D plots, contour plots, and heatmaps. Creating publication-quality graphs and exporting them to various formats. Animating graphs and visualizing time-series data. Final project: Students work on a real-world data analysis project and present their findings using Origin Software.

Core Compulsory Readings

1. Wes McKinney, Python for Data Analysis, O'Reilly, 2012, ISBN: 9781449319793
2. Claus O. Wilke, Fundamentals of Data Visualization: A Primer on Making, O'Reilly, 2019, ISBN: 9781492031086

Core Suggested Readings

1. Stephanie D. H. Evergreen, Effective Data Visualization The Right Chart for the Right Data, SAGE Publications, 2016, ISBN:9781506303079

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Skill Enhancement Course

MCCSA02SEC10 Quantum computing

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
2	0	2	2	0	2	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Quantum computing is a rapidly advancing field that promises to revolutionize computation and problem-solving. This course provides an introduction to quantum computing tailored specifically for non-physics students. It covers the fundamental principles of quantum mechanics and explores how quantum phenomena can be harnessed to perform powerful computations. No prior knowledge of physics or advanced mathematics is required

Course Objectives:

- Understand the basic principles of quantum mechanics.
- Explore the fundamental concepts of quantum computing.
- Gain knowledge of quantum algorithms and their applications.
- Develop an understanding of quantum gates and quantum circuits.

Learning Outcomes

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Develop a foundational understanding of quantum mechanics and its relevance to computing.
CO2	Gain knowledge of quantum algorithms and their potential applications.
CO3	Acquire practical skills in quantum programming and working with quantum circuits
CO4	Understand the challenges and future directions in quantum computing research

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓			✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Quantum Mechanics -Overview of classical computing and its limitations -Introduction to quantum mechanics -Wave-particle duality-Superposition and measurement-Quantum entanglement

Module 2: Quantum Computing Basics -Introduction to qubits and quantum gates -Quantum states and quantum operations - Quantum circuits and circuit model of Computation-Measurement and quantum measurement postulate

Module 3: Quantum Algorithms-Quantum parallelism and superposition - The Deutsch-Jozsa algorithm - Grover's algorithm for unstructured search -Shor's algorithm for factoring large numbers

Module 4: Quantum Simulators and Hardware- Overview of quantum simulators and their role in quantum computing research Introduction to quantum hardware (e.g., qubits, quantum gates) - Comparison of different quantum computing technologies (e.g., superconducting qubits, trapped ions) - **Quantum Programming Languages and Frameworks-** Introduction to quantum programming languages (e.g., Q#, Qiskit) - Basics of quantum program structure and execution - Quantum gates and operations in programming languages - Hands-on exercises using quantum programming frameworks

Core Compulsory Readings

Quantum Computing for Everyone" Author: Chris Bernhardt Publisher: The MIT Press Year: 2019

Core Suggested Readings

1. Programming Quantum Computers: Essential Algorithms and Code Samples" by Eric R. Johnston, Nic Harrigan, and Mercedes Gimeno-Segovia (O'Reilly Media)
2. Quantum Computing for Computer Scientists" by Noson S. Yanofsky and Mirco A. Mannucci (Cambridge University Press)
3. Quantum Computing: An Applied Approach" by Jack D. Hidary (Springer)

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

END OF SEMESTER II

Semester III

CORE COURSE

MSCSC03DSC14 MACHINE LEARNING TECHNIQUES

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4/1	0	5	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course is intended to familiarize the learners about the fundamental concepts in machine learning and deep learning mechanism for applying and analysing the huge volume of data generates in real life. The main purpose of this course is to provide foundational understanding of machine learning models and demonstrate how it will be useful for solving complex problems in real world.

Course Objectives:

- Aims to impart basic concepts of Artificial intelligence
- Impart knowledge on different learning scenario in machine learning
- Acquire the knowledge of various machine learning models and its implementation
- Understand various neural networks models architecture and its training algorithms
- Awareness of various dimensionality reduction techniques for optimizing the feature extraction techniques for improving the performance of various classification models
- Acquire the ability to analysis of data using various deep leaning techniques

Course Outcomes:

At the end of the course, the student will be able to:

SL #	Course Outcomes
CO1	Understand basic concepts of machine learning
CO2	Acquire knowledge on supervised and unsupervised learning
CO3	Obtain knowledge on the basics of neural networks
CO4	Understand concepts of deep learning architectures

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Machine Learning: Concept of learning task, inductive learning and the concepts of hypothesis space, introduction to different types of machine learning approaches, examples of machine learning applications, different types of learning: supervised learning, unsupervised learning, reinforcement learning. Setting up your machine learning platform: training, validation and testing, over-fitting and under-fitting, different types of error calculation.

Module 2: Supervised Learning: Introduction, learning a class from example, learning multiple classes, model selection and generalization: linear regression and feature selection, Bayesian and Decision Tree learning; classification tree and regression tree, multivariate methods for learning; multivariate classification and regression. Unsupervised Learning: Introduction, clustering; mixture densities, k-means clustering, expectation maximization algorithm, Latent Dirichlet Allocation, spectral and hierarchical clustering.

Module 3: Dimensionality reduction: principal component analysis, linear discriminant analysis, canonical correlation analysis. Introduction to Artificial Neural Network: Understanding brain, perceptron, Multi-Layer perceptron as universal approximator, general architecture of artificial neural network, feed forward and back propagation, different linear and nonlinear activation functions for binary and multi class classification.

Module 4: Introduction to Deep Learning: Fundamentals of deep learning, Deep Feedforward Networks, Regularization for Deep Learning, Optimization for Training Deep Models, Introduction to Convolutional Networks, Sequence Modelling using Recurrent Nets, overview of LSTM, fundamentals of Generative adversarial Network.

Core Compulsory Readings

1. EthemAlpaydin, Introduction to Machine Learning- 3rd Edition, PHI.

2. Ian Goodfellow and YoshuaBengio and Aaron Courville, Deep Learning (Adaptive Computation and Machine Learning), MIT Press, 2016.

Core Suggested Readings

1. Tom M. Mitchell, Machine Learning, McGraw-Hill

2. KuntalGanguly, Learning Generative Adversarial Networks, Packt Publishing, 2017.

TEACHING LEARNING STRATEGIES

Lecturing, case study/mini projects, Team Learning, presenting seminars on selected topics, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print.

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is Machine learning?
2. What are the different types of learning?
3. What is supervised learning?
4. What is reinforcement learning?
5. What is unsupervised learning?
6. What do you mean by training in Machine learning?
7. What is training dataset?
8. What do you mean by testing in Machine learning?
9. What is test data?

10. What is validation test?
11. What is meant by overfitting?
12. What do you mean by underfitting?
13. What is regression?
14. What is linear regression?
15. What is feature selection?
16. Explain classification tree.
17. Explain regression tree.
18. Explain about decision tree classifier.
19. What are the different multivariate methods for learning? Explain.
20. Explain the architecture of Artificial Neural Network.
21. Explain the training and testing algorithm for Backpropagation neural network.
22. Explain the different activation functions.
23. What is deep learning? Explain how it differs from machine learning.
24. What is LSTM?
25. What is CNN?

CORE COURSE

MSCSC03DSC15 COMPUTER GRAPHICS AND IMAGE PROCESSING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4/1	0	5	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course is designed to provide students with a comprehensive understanding of the principles, techniques and algorithms mainly used in Computer Graphics and in Image Processing. It explores the basic concepts and applications of creating, manipulating and rendering of digital images. It covers theoretical as well as practical implementation aspects of digital image processing.

Course Objectives:

- Give To introduce students to the basic concepts and principles of computer graphics and image processing.
- To develop skills in image processing techniques including image pre-processing, image seg9mentation, image enhancement and image filtering techniques.
- To understand the principles of modelling in computer graphics.
- To provide a hands-on experience and implementing image processing algorithms.
- To develop skills in creating and rendering 2D and 3D graphics.
- To understand the principles of geometric transformations and modelling in computer graphics.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	To understand the basics on computer graphics and transformation algorithms.
CO2	Learn different clipping algorithms on an image
CO3	To learn the operations on 2D digital images
CO4	Understand filtering and segmentation techniques on digital images.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓			✓	✓
CO2	✓	✓			✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Basic Concepts in Computer Graphics. Input devices. Display devices. Line and circle drawing Algorithms.Solid area scan-conversion.Polygon filling.Projections – Parallel, Perspective.Hidden Line Elimination Algorithms.Image processing – digital image representation – edge detection – Scene segmentation and labeling – region- labeling algorithm – perimeter measurement.Filled Area Primitives- Scan line polygon filling, Boundary filling and flood filling.Two dimensional transformations-Translation, Rotation, Scaling, Reflection and Shearing, Composite transformations, Matrix representations and homogeneous coordinates.Basic 3D transformations.

Module 2: Projections – Parallel and perspective projections – vanishing points. Visible surface detection methods– Back face removal- Z-Buffer algorithm, A-buffer algorithm, Depth-sorting method, Scan line algorithm. Window to viewport transformation. Cohen Sutherland Line clipping algorithm. Sutherland Hodgeman Polygon clipping algorithm. Three dimensional viewing pipelineVisible surface detection algorithms- Depth buffer algorithm.

Module 3: Introduction to Image processing and applications. Image as 2D data.Image representation in Gray scale, Binary and Colour images.Fundamental steps in image processing.Components of image processing system. Coordinate conventions. Sampling and quantization.Spatial and Gray Level Resolution.Basic relationship between pixels– neighbourhood, adjacency, connectivity.Fundamentals of spatial domain-convolution operation.

Module 4: Basics of spatial filtering - Smoothing spatial filterLinear and nonlinear filters, and sharpening spatial filters-Gradient and Laplacian. Fundamentals of Image Segmentation.Thresholding - Basics of Intensity thresholding and Global Thresholding. Region based Approach - Region Growing, Region Splitting and Merging. Edge Detection - Edge Operators- Sobel and Prewitt.

Core Compulsory Readings

1. Donald Hearn and M. Pauline Baker, Computer Graphics, PHI, 2e, 1996
2. E. Gose, R. Johnsonbaugh and S. Jost., Pattern Recognition and Image Analysis, PHIPTR, 1996 (Module IV – Image Processing part)
3. Computer Graphics: Principles and Practice", James D Foley, Andries van Dam, Steven K Feiner, John F Hughes
4. Digital Image Processing", Rafael C Gonalez and Richard E Woods

Core Suggested Readings

1. JZhigang Xiang and Roy Plastock, Computer Graphics (Schaum's outline Series), McGraw Hill, 1986.
2. William M. Newman and Robert F. Sproull , Principles of Interactive Computer Graphics. McGraw Hill, 2e, 1979
3. "Digital Image Processing: Concepts, Algorithms and Scintific Applications", BerdJahne

TEACHING LEARNING STRATEGIES

- Lecturing, Interactive discussions, hands-on exercises, lab sessions, collaborative learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion Lecture, Seminar, Discussion, audio and video presentation, demonstration, practical assignments and exercises

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Explain the concept of anti-aliasing and the techniques used in Computer Graphics. How it will be affected in the quality of an image.
2. Discuss the concept of hidden surface removal in Computer Graphics processing. Explain Depth Buffer algorithm along with its advantages as well as limitations.
3. What are the different steps involved in ray tracing? Explain its advantages and ligations.

4. Explain how computer vision is related to Computer Graphics. Discuss the challenges of computer vision in real world scenarios.
5. What is the role of color models in Computer Graphics? Discuss RGB and CMYK color models and their application in the contest of real-world scenario.
6. Explain different image segmentation techniques. Also discuss about image segmentation evaluation metrics.
7. Explain image enhancement in image processing. Discuss different techniques used for image enhancement.
8. Explain basic steps involved in image processing by explaining each. Discuss the importance of each one.
9. Explain the concept of image registration in image processing. Explain how noise removal and image deblurring affects the overall performance of a digital image.
10. What is image compression in image processing. Differentiate lossless and lossy compression and its importance.

CORE COURSE

MSCSC03DSC16 QUANTUM COMPUTING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3/1	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Quantum computing is a rapidly advancing field that promises to revolutionize computation and problem-solving. This course provides an introduction to quantum computing tailored specifically for non-physics students. It covers the fundamental principles of quantum mechanics and explores how quantum phenomena can be harnessed to perform powerful computations. No prior knowledge of physics or advanced mathematics is required

Course Objectives:

- Understand the basic principles of quantum mechanics.
- Explore the fundamental concepts of quantum computing.
- Gain knowledge of quantum algorithms and their applications.
- Develop an understanding of quantum gates and quantum circuits.
- Explore the challenges and potential of quantum computing.
- Gain familiarity with quantum programming languages and frameworks.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Develop a foundational understanding of quantum mechanics and its relevance to computing.
CO2	Gain knowledge of quantum algorithms and their potential applications.
CO3	Acquire practical skills in quantum programming and working with quantum circuits.
CO4	Understand the challenges and future directions in quantum computing research.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2		✓		✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓		✓	✓	

COURSE CONTENTS

Module 1: Introduction to Quantum Mechanics -Overview of classical computing and its limitations - Introduction to quantum mechanics -Wave-particle duality-Superposition and measurement-Quantum entanglement

Module 2: Quantum Computing Basics -Introduction to qubits and quantum gates -Quantum states and quantum operations - Quantum circuits and circuit model of Computation-Measurement and quantum measurement postulate

Module 3: Quantum Algorithms-Quantum parallelism and superposition - The Deutsch-Jozsa algorithm - Grover's algorithm for unstructured search -Shor's algorithm for factoring large numbers

Module 4: Quantum Simulators and Hardware- Overview of quantum simulators and their role in quantum computing research Introduction to quantum hardware (e.g., qubits, quantum gates) - Comparison of different quantum computing technologies (e.g., superconducting qubits, trapped ions) - **Quantum Programming Languages and Frameworks-** Introduction to quantum programming languages (e.g., Q#, Qiskit) - Basics of quantum program structure and execution - Quantum gates and operations in programming languages - Hands-on exercises using quantum programming frameworks

Module X (For Additional Reading and Comprehension by the Students):

Advanced Quantum Computing

- Quantum Fourier transform and its applications
- Shor's factorization algorithm and quantum cryptography
- Quantum simulation and quantum machine learning

- Introduction to quantum error correction

Quantum Computing Applications

- Quantum computing in cryptography and secure communication
- Quantum optimization and quantum annealing
- Quantum simulation of physical systems
- Potential impact of quantum computing on various industries

Programming Quantum Computers

- Introduction to quantum software development kits (SDKs)
- Quantum programming languages: Qiskit, Cirq, and PyQuil
- Building and running quantum circuits

Core Compulsory Readings

Quantum Computing for Everyone" Author: Chris Bernhardt Publisher: The MIT Press Year: 2019

Core Suggested Readings

1. Programming Quantum Computers: Essential Algorithms and Code Samples" by Eric R. Johnston, NicHarrigan, and Mercedes Gimeno-Segovia (O'Reilly Media)
2. Quantum Computing for Computer Scientists" by Noson S. Yanofsky and Mirco A. Mannucci (Cambridge University Press)
3. Quantum Computing: An Applied Approach" by Jack D. Hidary (Springer)

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion and lab

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is a qubit in quantum computing and how is it different from a classical bit?
2. What is superposition in quantum computing and why is it important for quantum algorithms?
3. Explain the concept of entanglement in quantum computing and its significance in quantum communication and computation.
4. Describe the fundamental difference between quantum gates and classical logic gates, and provide an example of a commonly used quantum gate.
5. Compare and contrast quantum computing with classical computing. Highlight at least three key differences between the two.
6. Explain the concept of quantum algorithms and provide an overview of an important quantum algorithm, such as Shor's algorithm or Grover's algorithm.

ELECTIVE COURSE - POOL B

MSCSC03DSE04 Foundations of Data Science

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3	0	3	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

Foundations of Data Science is an introductory course designed to provide students with a strong grounding in the fundamental concepts, techniques, and tools used in the field of data science. The course covers various aspects of data science, including data collection, data cleaning, data analysis, and data visualization. Through a combination of lectures, hands-on projects, and practical exercises, students will develop the essential skills required to work with data effectively and derive meaningful insights from it.

Course Objectives:

- To introduce students to the field of data science and its applications in diverse industries.
- To familiarize students with the data science workflow and the steps involved in processing and analyzing data.
- To equip students with essential programming skills using languages such as Python and R.
- To provide a comprehensive understanding of statistical concepts and their applications in data analysis.
- To introduce students to machine learning algorithms and their use in predictive modeling and clustering tasks.
- To emphasize the importance of ethical considerations in data science and responsible data handling practices.

Course Outcomes:

By the end of this course, students should be able to:

SL #	Course Outcomes
CO1	Proficiently apply programming languages (Python and R) for data manipulation, analysis, and machine learning.
CO2	Utilize statistical techniques to explore data, make data-driven decisions, and create meaningful visualizations.
CO3	Demonstrate an understanding of the data science workflow, including ethical considerations in data collection and privacy.
CO4	Collaborate effectively on data science projects and communicate results to diverse audiences.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓	✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3				✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

Course Contents

Module 1: Introduction to Data Science .Overview of Data Science and its applications in various industries. Understanding the Data Science workflow: data collection, data cleaning, data exploration, data analysis, and data visualization. Introduction to data types, data formats, and data storage. Introduction to programming languages and libraries commonly used in data science (e.g., Python, R, NumPy, Pandas).Basics of data manipulation and cleaning techniques.

Module 2: Statistics and Data Analysis .Introduction to descriptive and inferential statistics. Measures of central tendency and variability. Hypothesis testing and confidence intervals. Correlation and regression analysis. Understanding probability distributions (e.g., normal, binomial, Poisson) and their applications in data science. Data visualization techniques: scatter plots, histograms, box plots, etc.

Module 3: Data Processing and Machine Learning . Data preprocessing techniques: data scaling, encoding categorical variables, handling missing data, and feature engineering. Introduction to supervised, unsupervised, and semi-supervised machine learning algorithms. Linear and logistic regression for predictive modeling. Decision trees, random forests, and ensemble methods. Clustering algorithms (e.g., k-means, hierarchical clustering) for unsupervised learning. Model evaluation and performance metrics.

Module 4: Big Data and Data Ethics. Introduction to Big Data concepts and challenges in processing large datasets. Introduction to distributed computing and tools like Apache Hadoop and Spark. Ethical considerations in data science: privacy, bias, and fairness in data collection and analysis. Responsible data handling practices and data governance.Case studies and real-world applications in various domains.

Book of Study:

1. "Foundations of Data Science: An Introduction to Data Analysis and Machine Learning" Author: John Doe Publisher: XYZ Press Year: 2023
2. Reference Book: Title: "Python for Data Analysis" Author: Wes McKinney Publisher: O'Reilly Media Year: 2021
3. Reference Book: Title: "Introduction to Statistical Learning" Authors: Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani Publisher: Springer Year: 2017
4. Reference Book: Title: "Data Science for Business" Authors: Foster Provost and Tom Fawcett Publisher: O'Reilly Media Year: 2013

ELECTIVE COURSE - POOL B

MSCSC03DSE05 FOUNDATIONS OF NATURAL LANGUAGE PROCESSING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3	0	3	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course introduces the fundamentals of Natural Language Processing (NLP) from an algorithmic viewpoint. The course provides insight into how machines can deal with NLP. A gist about the various applications of NLP is also discussed.

Course Objectives:

- To introduce the fundamentals of NLP from an algorithmic viewpoint
- To introduce the use of CFG and PCFG in NLP
- To illustrate the process of syntax analysis in NLP
- To explain the fundamentals of speech processing in NLP

Course outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire basic knowledge on natural language and automata.
CO2	Illustrate the process of syntax analysis in NLP
CO3	Apply grammars and parsing in NLP.
CO4	Discuss some applications of Natural Language Processing (NLP)

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4			✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Language: Linguistic Knowledge, Grammar, Language and Thought, computational linguistics vs NLP, why NLP is hard ?, why is NLP useful ?, classical problems. Words of Language, Content Words and Function Words, Lexical categories, Regular expressions and automata. Morphology: Morphemes, Rules of Word Formation, Morphological parsing and Finite state transducers.

Module 2: N-grams: simple N-grams, smoothing, Applications, language modeling. Word classes and POS tagging: tag sets, techniques: rule based, stochastic and transformation based. Introduction to Natural Language Understanding - Levels of language analysis - Syntax, Semantics, Pragmatics.

Module 3: Grammars and Parsing - Grammars for Natural Language: CFG, Probabilistic Context Free Grammar, Statistical Parsing. Features and Unification: Feature Structures and Unification of feature structures. Lexical semantics, formal semantics and discourse. WSD, Information retrieval: Boolean, vector space and statistical models. Knowledge Representation and Reasoning - FOPC, Elements of FOPC.

Module 4: Discourse processing: monologue, dialogue, reference resolution, Conversational Agent. Text coherence. Dialogue acts: Interpretation of dialogue acts, plan inference model, clue-based model. Semantics: Representing meaning, Semantic analysis, Lexical semantics. Applications: Machine Translation, Natural Language Generation: architecture, surface realization and discourse planning.

Core Compulsory Readings

Daniel Jurafsky and James H Martin.Speech and Language Processing.

Core Suggested Readings

1. Hobson Lane, Cole Howard, HannesHapke. Natural Language Processing in Action
2. Victoria fromkin, Robert Rodman and Nina Hyams, An Introduction to language, Tenth Edition. Downloadable freely at: https://ukhtt3nee.files.wordpress.com/2019/04/an_introduction_to_language.pdf

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. List the reasons for choosing CFG to represent language in parsing

2. Distinguish between (i) semantics, pragmatics and discourse

3. (i) Write different types of inferences.

(ii) Write FOPC for the following sentences:

All cats and dogs hate each other

I arrived in New York

ELECTIVE COURSE - POOL B

MSCSC03DSE06 SPEECH AUDIO AND VIDEO FORENSICS

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3	0	3	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course "Speech, Audio, and Video Forensics" offers students an in-depth exploration of the techniques and methodologies used in the analysis and investigation of speech, audio, and video evidence in legal and forensic settings. Students will learn how to identify and authenticate audio and video recordings, analyse speech patterns, and enhance and restore audio and video quality. The course covers various aspects, including forensic speaker identification, voice biometrics, audio and video tampering detection, and analysis of surveillance footage. Through hands-on exercises and case studies, students will develop practical skills in forensic analysis and gain a comprehensive understanding of the field of speech, audio, and video forensics.

Course Objectives:

- Understand the fundamental concepts and principles of speech, audio, and video forensics.
- Learn techniques for analyzing and authenticating speech, audio, and video evidence.
- Develop skills in enhancing and restoring audio and video recordings for forensic purposes.
- Gain knowledge of forensic speaker identification and voice biometrics.
- Acquire proficiency in detecting tampering and analyzing surveillance footage through practical exercises and case studies.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Obtain knowledge on speech production system.
CO2	Obtain basic knowledge on Forensic Linguistics.
CO3	Acquire basic knowledge in Forensic speaker recognition.
CO4	Obtain fundamental knowledge in Audio /video forensics

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1				✓	✓	✓
CO2		✓		✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Physics of sound: waves and sound, analysis and synthesis of complex waves, Human and nonhuman utterances, anatomy of vocal tract, vocal formants, analysis of vocal sound, frequencies and overtones. Electronics of Audio Recording, Transmission and Playback devices, noise and distortion, voice storage and preservation

Module 2: Forensic Linguistics: Phonetics, Morphology, Syntax, Semantics, Stylistics, Pragmatics, Script, orthography and graphology, Difference between language and speech, Psycholinguistics, Neurolinguistics, Sociolinguistics, Scientific approaches; Reliability and admissibility of evidence in the court, linguistic profile, language register Discourse Analysis: Connivance, acceptance, listening feedback and rejection in the context of Mens-Rea, Narrative, Dialectology, Linguistic variety as a geographical marker, Idiolects and speaker characterization, Phonology, Morphology and Word formation processes as individual linguistic abilities.

Module 3: Various approaches in Forensic Speaker Identification, Instrumental Analysis of speech sample, Interpretation of result, Statistical interpretation of probability scale, Objective/Subjective methods, discriminating tests, closed test, open test, likelihood ratio calculation, Concept of test and error in Speaker Identification, ase studies. Techniques and Best Practices for examination of Audio recording authentication and case studies.

Module 4: Audio /video forensics: Spectrography – Conversion of different voice file formats in to forensic voice module formats. Various types of spectrograms, spectrographic cues for vowels and consonants.Speech analysis in forensic sciences.Speech synthesis by analysis, Speech recognition and speaker identification.Fundamentals of Digital Signal processing and communication system. Analogue and digital systems, Analogue signal and digital signals, Analogue to digital and digital to analogue converters, need and advantages of digital systems and digital signal processing. Forensic extraction of video files from DVR and other storage media.Forensic examination of DVR containing video footages, its frame analysis. Forensic examination and authentication of meta data present in video/audio files. Enhancement of video/ Photo and its comparison/authentication.

Core Compulsory Readings

1. Bengold& Nelson Moryson; “Speech and Audio signal processing”, John Wiley & Sons, USA (1999)
2. D.B. Fry; “The Physics of Speech, Cambridge University Press”, (2004)

Core Suggested Readings

1. Dwight Bolingeret. al.; “Aspects of Language”, Third Edition, Harcourt Brace Jovanovich College Publishers, USA, (1981)
2. Gloria J. Borden et. al.; “Speech Science Primer (Physiology, Acoustics and perception of Speech)”, 6th Ed, a Wolters Kluwer Company, USA, (2011)
3. Harry Hollien; “Forensic Voice Identification”, Academic Press, London. (2001)
4. Harry Hollien; “The Acoustics of Crime- The New Science of Forensic Phonetics”, Plenum Press, New York and London (1990)
5. Oscar Tosi; “Voice Identification-Theory of Legal Applications”, University Park Press, Baltimore (1979)
6. O'Shaughnessy, Douglas; “Speech Communication”, Hyderabad Universities Press (India) Pvt. Ltd. (2001)
7. Patricia Ashby; “Speech Sounds”, 2nd Ed. Routledge, London and New York (2005)
8.Philip Rose; “Forensic Speaker Identification,” Taylor and Francis, Forensic Science Series, London (2002)
9.Simon J. Godsill; “Digital Audio Restoration”, Springer, (1998)

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What are the key principles and techniques used in speech, audio, and video forensics?
2. How can speech patterns and characteristics be analyzed to identify individuals in forensic investigations?
3. What are the methods and tools available for authenticating audio and video recordings?
4. How can audio and video recordings be enhanced and restored for better clarity and intelligibility?
5. What are the techniques used in forensic speaker identification and voice biometrics?
6. How can tampering and manipulation of audio and video evidence be detected in forensic analysis?
7. What are the challenges and considerations in analyzing surveillance footage for forensic purposes?
8. What are the legal and ethical considerations in speech, audio, and video forensics?
9. How can digital forensics play a role in speech, audio, and video investigations?
10. What are the real-world applications and case studies of speech, audio, and video forensics in criminal investigations or legal proceedings?

ELECTIVE COURSE - POOL B

MSCSC03DSE07 INTERNET OF THINGS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3	0	3	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course on Internet of Things (IoT) provides an in-depth understanding of the interconnected network of physical devices and their ability to communicate and exchange data. It explores the various technologies, protocols, and architectures that enable IoT systems to function effectively. Students will learn about sensor technologies, data analytics, and cloud computing, which are essential components of IoT ecosystems. The course also covers the challenges and security considerations associated with IoT implementations, as well as the potential impact of IoT on industries such as healthcare, manufacturing, and smart cities. By the end of the course, students will be equipped with the knowledge and skills to design, develop, and manage IoT solutions and leverage the transformative potential of this rapidly evolving field.

Course Objectives:

- Understand the fundamental concepts and principles of the Internet of Things (IoT) and its applications.
- Gain knowledge of the different technologies, protocols, and architectures used in IoT systems.
- Develop skills to design and implement IoT solutions using sensor technologies, data analytics, and cloud computing.
- Recognize the challenges and security considerations associated with IoT deployments and learn strategies to mitigate them.
- Explore the potential impact of IoT in various industries and understand its role in transforming business processes and enabling smart environments.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the basic of IoT .
CO2	Study IoT Architecture models
CO3	Study various communication protocols for IoT.
CO4	Design a PoC of an IoT system using Rasperry Pi/Arduino

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓			✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to IoT: Internet of Things - Physical Design- Logical Design- IoT Enabling Technologies - IoT Levels & Deployment Templates - Domain Specific IoTs - IoT and M2M - IoT System Management with NETCONF-YANG- IoT Platforms Design Methodology.

Module 2: IoT Architecture: M2M high-level ETSI architecture - IETF architecture for IoT - OGC architecture - IoT reference model - Domain model - information model - functional model - communication model - IoT reference architecture. IoT Protocols: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Unified Data Standards.

Module 3: Protocols – IEEE 802.15.4 – BACnet Protocol – Modbus– Zigbee Architecture – Network layer – 6LowPAN - CoAP–Security. Building IoT with RASPBERRY PI &ARDUINO:Building IOT with RASPERRY PI- IoT Systems - Logical Design using Python – IoT Physical Devices & Endpoints - IoT Device -Building blocks -Raspberry Pi -Board - Linux on Raspberry Pi - Raspberry Pi Interfaces.

Module 4: Programming Raspberry Pi with Python - Other IoT Platforms - Arduino. Case Studies and Real World Applications :Real world design constraints - Applications - Asset management, Industrial automation, smart grid, Commercial building automation, Smart cities - participatory sensing - Data Analytics for IoT – Software & Management Tools for IoT Cloud Storage Models & Communication APIs - Cloud for IoT - Amazon Web Services for IoT.

Core Compulsory Readings

1. ArshdeepBahga, Vijay Madisetti, ―Internet of Things – A hands-on approach‖, Universities Press, 2015
2. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), ―Architecting the Internet of Things‖, Springer, 2011.
3. Honbo Zhou, ―The Internet of Things in the Cloud: A Middleware Perspective‖, CRC Press, 2012.

Core Suggested Readings

1. 4. Jan Ho¨ ller, VlasiosTsiatsis , Catherine Mulligan, Stamatis , Karnouskos, Stefan Avesand. David Boyle, "From Machine-to-Machine to the Internet of Things - Introduction to a New Age of Intelligence", Elsevier, 2014.
2. 5. Olivier Hersent, David Boswarthick, Omar Elloumi , ―The Internet of Things – Key applications and Protocols‖, Wiley, 2012

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is the Internet of Things (IoT) and how does it differ from traditional Internet connectivity?
2. What are the main components that make up an IoT system?
3. How do IoT devices communicate with each other and with cloud platforms?
4. What are some common applications of IoT in industries such as healthcare, agriculture, and transportation?
5. What are the key security challenges and considerations when deploying IoT solutions?
6. How does data analytics play a role in extracting insights from the vast amount of data generated by IoT devices?
7. What are the different connectivity protocols used in IoT and how do they impact device interoperability?
8. How can edge computing enhance the performance and efficiency of IoT systems?
9. What are the ethical and privacy implications of collecting and analyzing data from IoT devices?
10. What is the future outlook for IoT and what potential advancements and innovations can we expect to see in the coming years?

ELECTIVE COURSE - POOL B

MSCSC03DSE08 PATTERN RECOGNITION

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3	0	3	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course is designed to provide the basics and various techniques of pattern recognition. The course provides an insight to various clustering algorithms. A discussion about soft computing techniques for Pattern Recognition.

Course Objectives:

- To understand the basics of Pattern Recognition
- To familiarize with various techniques of Pattern Recognition
- To understand the various algorithms for clustering
- To create an awareness about soft computing techniques for Pattern Recognition

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the fundamentals of Pattern Recognition.
CO2	Illustrate various algorithms for classification and clustering.
CO3	Acquire knowledge on feature selection and extraction methods.
CO4	Understand the concepts of soft computing techniques for Pattern Recognition.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓	✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Pattern recognition systems - Definitions, data representation, representations of patterns and classes. Types of pattern recognition systems.Applications of pattern recognition systems.Bayesian decision making and Bayes Classifier for continuous and discrete features.

Module 2: Min - max and Neymann - Pearson classifiers, Discriminant functions, Decision surfaces. Maximum likelihood estimation and Bayesian parameter estimation. Overview of Nonparametric density estimation - Histogram based approach, classification using Parzen window. K - nearestneighbor estimation and classification. Classification of clustering algorithm - hierarchical clustering - agglomerative clustering.Partitional clustering Forgy’s algorithm.K-means clustering.

Module 3: Introduction to feature selection – filter method - sequential forward and backward selection algorithms. Wrappers method and embedded methods. Feature extraction methods - Principal component analysis, fisher linear discriminant analysis, ICA.

Module 4: Neural network structures for Pattern Recognition – Neural network based Pattern associators – Unsupervised learning in neural Pattern Recognition – Self Organizing networks – Fuzzy logic – Fuzzy pattern classifiers – Pattern classification using Genetic Algorithms.

Core Compulsory Readings

1. Duda R.O., and Har P.E., Pattern Classification and Scene Analysis, Wiley, New York, 1973.

Core Suggested Readings

1. Bishop C. M, Pattern recognition and machine learning, Springer, 2nd Edition, 2006
2. Theodoridis .S, Pikrakis .A, Koutroumbas .K, Cavouras .D, Introduction to Pattern Recognition: AMatlab approach, Academics Press 2010

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1**.** What is pattern recognition, and how does it differ from other forms of data analysis?

2. Describe the main steps involved in the pattern recognition process.

3. How are feature extraction methods used to represent patterns in a more meaningful way?

4. Explain the concept of supervised learning in pattern recognition and provide an example application.

5. What are the key differences between parametric and non-parametric pattern recognition algorithms?

6. Discuss the challenges and potential solutions for handling high-dimensional data in pattern recognition.

7. How do support vector machines (SVMs) work, and what are their advantages in pattern recognition tasks?

8. Can you elaborate on the concept of unsupervised learning in pattern recognition and its practical applications?

9. What role does dimensionality reduction play in pattern recognition, and how does it help improve performance?

10. How are neural networks and deep learning architectures used in pattern recognition, and what are their benefits compared to traditional methods?

ELECTIVE COURSE - POOL B

MSCSC03DSE09 COMPUTER VISION

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
3	0	3	3	0	3	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The program focuses on building a broad understanding on the fundamentals of computer vision. The objective of the course is to communicate theoretical knowledge about machine vision. This is a field of artificial intelligence that enables computers and systems to derive meaningful information from visual inputs and has the prospective to energies scientific advancement through technological revolution.

Course Objectives:

- To introduce and discuss the basic concepts of computer vision techniques
- Understandings on the applications of machine vision
- To accomplish knowledge about motion analysis

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Awareness about the importance of computer vision.
CO2	Familiar with object representation
CO3	Acquire the knowledge about 3D vision
CO4	Gain information about the applications of motion analysis techniques.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction: Motivation, Difficulty, Image analysis tasks, Image representations, Image digitization, Image properties, Color images, Cameras. Data Structures: Levels of image data representation - Traditional image data structures - Hierarchical data structures. Texture: Statistical texture description, Syntactic texture description methods, Hybrid texture description methods, Texture recognition method applications.

Module 2: Object Recognition: Knowledge representation, Statistical pattern recognition, Neural nets, Syntactic pattern recognition, Recognition as graph matching, Optimization techniques in recognition, Fuzzy systems.

Module 3: 3D vision: 3D vision: Tasks - Basics of projective geometry - Scene construction from multiple views, Uses: Shape from X - Full 3D objects - 3D model based vision - 2D view based 3D representation.

Module 4: Motion Analysis: Differential motion analysis methods, Optical flow, Analysis based on interest points, Detection of specific motion patterns, Video Tracking, Motion models to aid tracking.

Core Compulsory Readings

1. Szeliski, Richard, Computer Vision: Algorithms and Applications- 2ndEdn, Springer-
2. Verlag,2022.
3. Milan Sonka, Vaclav Hlavac and Roger Boyle, "Image Processing, Analysis and Machine Vision", Cengage Learning, New Delhi, 2014.
4. Wesley E. Synder and Hairong Qi, "Machine Vision", Cambridge University Press, USA, 2010.

Core Suggested Readings

1. Rafael C Gonzalez, Richard E Woods, Steven L Eddins, "Digital Image Processing",
2. Pearson Education, New Delhi, 2009.
3. Shapiro L and Stockman G, Computer Vision, Prentice-Hall, 2001.

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. How do humans recognize objects and how can we replicate this process in machines?
2. What are some applications of projective geometry in 3D vision, and how do they impact industries such as robotics, autonomous vehicles, and augmented reality?
3. What are some challenges faced when working with 2D view based 3D representations?
4. How can motion models be adapted to different tracking scenarios, such as in crowded environments, fast-moving objects, or occluded scenes?
5. What are the different types of video tracking algorithms, and how do they differ in terms of accuracy, speed, and robustness?
6. How can machine learning be used to improve the accuracy and efficiency of motion pattern detection in computer vision? Discuss the advantages and limitations of deep learning, supervised learning, and unsupervised learning methods in this context.

7. What are the main applications of motion pattern detection in computer vision, and how can they be applied in real-world scenarios? Discuss the challenges associated with deploying these methods in real-world applications.

8. What are the potential future directions of motion models in tracking, such as in 3D tracking or dynamic scene analysis? Discuss the need for novel algorithms and hardware advancements to realize these directions.

9. What is the role of camera calibration in multi-view stereo reconstruction, and how can it be performed accurately?

10. How can optical flow be used for motion analysis and object tracking, and what are the main challenges in these tasks?

S3 – Multi-Disciplinary Courses

Multi-Disciplinary Course

MSCSC03MDC01 DESIGN AND ANALYSIS OF ALGORITHMS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The objective of course is to impart theoretical knowledge in the specialized area of algorithm design and analysis. Study of algorithms is very substantial in classification of problems and their solutions based on complexity. Analysis of algorithms provides a means for choosing an appropriate algorithm for solving a problem at hand. The course provides an insight into all aspects of computational complexity and the use, design, analysis and experimentation of efficient algorithms. The better understanding paves way for successful implementations in various scientific applications. The course will focus on various advanced paradigms and approaches used to design and analyses algorithms.

Course Objectives:

- To introduce basic principles that drive various algorithm design strategies
- Discuss the complexity analysis techniques and overview of P, NP problems
- Discuss about the concept of design and analysis of parallel algorithms

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Accomplish Knowledge about important computational problems and acquire knowledge to design the algorithm.
CO2	Obtain knowledge to analyze algorithm control structures and solving recurrence.
CO3	Attain information about Complexity Classes
CO4	Accomplish knowledge about Parallel Algorithms

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓	✓	✓		✓	
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Algorithm Design: Introduction, Steps in developing algorithm, Methods of specifying an algorithm, Decisions prior to designing: based on the capabilities of the device, based on the nature of solutions, based on the most suitable data structures. Important Problem Types: Sorting, Searching, String processing, Graph problems, Combinatorial problems, Geometric problems and Numerical problems. Basic Technique for Design of Efficient Algorithm: Brute Force approach (String matching), Divide-and-Conquer approach (Merge sort), Branch-and-Bound technique (Knapsack problem). Greedy approach (Kruskal's algorithm and Prim's Algorithm), Dynamic Programming (Longest Common Subsequence), Backtracking (Sum of subsets problem).

Module 2: Algorithm Analysis: Importance of algorithm analysis, Time and Space Complexity. Growth of Functions: Asymptotic notations, Cost estimation based on key operations- Big Oh, Big Omega, Little Oh, Little Omega and Theta notations, Big Oh Ratio Theorem, Big Theta Ratio Theorem, Big Omega Ratio Theorem. Analyzing Algorithm Control Structures, Solving Recurrences: Iteration Method, Substitution Method, The Recursion Tree Method, Master's Theorem, Problem solving using Master's Theorem Case 1, Case 2 and Case 3. Analysis of Strasser's algorithm for matrix multiplication, Analysis of Merge sort.

Module 3: Complexity - Complexity Classes: P, NP, NP Hard and NP Complete problems. NP Completeness reductions for Travelling Salesman Problem and Hamiltonian Cycle.P versus NP problem.

Module 4: Design and Analysis of Parallel Algorithms: PRAM models – EREW, ERCW, CREW and CRCW, Relation between various models, Handling read and write conflicts, work efficiency, Brent's theorem. Analyzing Parallel Algorithms: Time Complexity, Cost, Number of Processors, Space Complexity, Speed up, Efficiency, Scalability, Amdahl's Law. Euler Tour Technique, Parallel prefix computation, Parallel merging and sorting.

Core Compulsory Readings

1. Thomas H Cormen, Charles E Leiserson, and Ronald L Rivest, Introduction to Algorithms, 3rd Edition, Prentice Hall of India Private Limited, New Delhi

2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms, Addison Wesley

3. Pallaw, V K, Design and Analysis of Algorithms, Asian Books Private Ltd, 2012.

4. Razdan S, Fundamentals of Parallel Computing, Narosa Publishing House, 2014.

Core Suggested Readings

1. Pandey H M, Design and Analysis of Algorithms, University Science Press, 2013

2. Upadhyay, N, Design and Analysis of Algorithms, SkKataria& Sons, 2008.

3. U. Manber, Introduction to Algorithms: A Creative Approach, Addison Wesley,

4. Gilles Brassard and Paul Bratley, Fundamentals of Algorithmics, Prentice-Hall of India

5. Goodman S E and Hedetniemi, Introduction to the Design and Analysis of Algorithms, Mcgraw Hill

6.Horowitz E and Sahni S, Fundamentals of Computer Algorithms, Galgotia Publications Pvt. Ltd

7.OdedGoldreich, P,NP and NP- Completeness, Cambridge University Press, 2011.

8. Donald Knuth, The Art of Computer Programming, Fundamental Algorithms, Volume - 1, Addison Wesley, 1997.

9. SanjeevArora and Boaz Borak, Computational Complexity- A Modern Approach, Cambridge University Press; 2009.

10. Daniel Hills W and Bruce M Boghosian, Parallel Scientific Computation, Science, Vol 261, Pp. 856-863

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Differentiate Direct Recursion and Indirect Recursion.

2. Solve the recurrence $\mathbf{T(n) = 2T(\sqrt{n}) + \log n\ 2}$.

3. Compute the time complexity of the following:

```
for i ← 100 to m-1
{ for j ← 10 to i
   {
      A ← B + C[i][j]
    }
 }
```

4. Let $\mathbf{T(n) = 4\,T(n/2) + n^3}$, then show that $\mathbf{f(n) = \Omega(n^3)}$ and $\mathbf{T(n) = \theta(n^3)}$.

5.Prove that Hamiltonian Cycle is NP Complete.

6. How can we solve Knapsack problem using Branch-and-Bound technique?

7. Given a set S = **{2, 4, 6}** and Weight = **6**. Find subset sum using backtracking approach.

8. Let **H(t)** be the number of multiplications in the following:

```
int Factorial( intt)
 {
if (t = = 0)
then return 1
else
returnt* Factorial(t-1)
  }
```

Prove that **H(t) = t.**

9. 'The running time is directly proportional to the frequency count of the algorithm.' Explain the meaning of the statement in detail.

10. The recurrence $\mathbf{T(n) = 7T(n/2) + n^2}$describes the running time of an algorithm **A**. A competing algorithm $\mathbf{A^I}$ has a running time of $\mathbf{T^I(n)= kT^I(n/4)+n^2}$**.** What is the largest integer value for **k** such that $\mathbf{A^I}$ is asymptotically faster than **A**?

11. What do mean by Parallel Prefix Computation?

12. Can the master theorem be applied to the recurrence $\mathbf{T(n) = 2T(n/2) + n \log n}$ **?** Why or Why not?

Multi-Disciplinary Course

MSCSC03MDC02 PRINCIPLES OF PROGRAMMING AND NUMERICAL METHODS

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course provides an insight into building programs using Python programming language, especially for numerical methods. Various numerical methods and concepts of differentiation and integration are also discussed.

Course Objectives:

- To understand various numerical methods
- To acquire knowledge about Errors and Approximations
- To develop programs using numerical methods
- To understand the basic concepts of Python programming

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Acquainted with Numerical Methods.
CO2	Obtain the knowledge about Numerical Integration, Differentiation and its applications.
CO3	Understanding the basic concepts of Python programming.
CO4	Ripen skill in programming.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓			✓	
CO2	✓	✓			✓	
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Programming: Basic concepts, algorithm, flow chart, programming languages – classification. Syntax and semantics of programming languages - Imperative and OO Languages. Functional Languages - Logic Programming Languages. Features of Python, Different Methods to Run Python, Basic Elements (Objects, Expressions, Numerical Types, Strings, Variables), Comments, Indentation in Python, Input and Output in Python, import function, Operators in Python, Branching (if, else, elif), Iteration (while, for), range and enumerate functions, Tuples, Lists, Sets, Dictionaries, Built - in methods of lists, sets and dictionaries, Mutable and Immutable Objects.

Module 2: Functions Definition, Function Calling, Function Arguments (Required, Keyword, Default), Recursion, Modules, Built-in Modules, Creating Modules, File Handling (Opening, Closing, Writing, Reading), Exceptions, Built-in Exceptions (IndexError, OverflowError, ZeroDivisionError, RuntimeError), Exception Handling. Class Definition, Object Creation, Built-in Attribute Methods, Object Oriented Programming Features of Python. Arrays in Python, Numpy Module, ndarray, Creating Arrays (array, zeros, ones, empty, linspace, arrange, random), Two-Dimensional Array, Indexing, Slicing, Iterating, Copying, Splitting, Shape Manipulation (reshape, transpose, resize), Arithmetic Operations on Arrays. Data Visualization in Python matplotlib Module, pyplot, plot(), scatter, bar charts, Formatting, figure(), subplot(), text(), xlabel(), ylabel(), title(), Plotting Simple Mathematical Functions (sin x, x^2).

Module 3: Introduction to Numerical Methods: Nature of numerical problems; computer based solutions. Errors and Approximations. Nonlinear equations – Bisection Method, Regular - Falsie Method, Newton Raphson. System of Linear Equations - Gauss elimination, Gauss Jordan elimination, Triangulation method, Iterative method, Jacobi. Case study by writing algorithms.

Module 4: Numerical Integration and Differentiation: Concept of differentiation and Integration. Taylors series and Eulers methods-Simpson's Romberg, Gaussian, Runge - Kutta methods. Case study by writing algorithms.

Core Compulsory Readings

1. Jean-Paul Tremblay, R Manohar, Discrete Mathematical Structures with Application to Computer Science, McGraw Hill
2. Sastry S.S., Introductory Methods of Numerical Analysis, January 2012 PHI
3. Balagurusamy, E., Numerical Methods, Tata McGraw-Hill, New Delhi, 1999
4. R. G. Dromey, How to solve it by computer, Pearson education, fifth edition, 2007
5. Dr.Jeeva Jose, Taming Python By Programming, Khanna Publishing
6. John V. Guttag, Introduction to Computation and Programming Using Python with Application to Understanding Data, PHI (2016)
7. https://www.numpy.org/devdocs/user/quickstart.html
8. https://matplotlib.org/2.0.2/users/pyplot_tutorial.html

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Multi-Disciplinary Course

MSCSC03MDC03 JAVA PROGRAMMING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course introduces JAVA programming language with object-oriented programming principles. Emphasis is placed on developing skills on implementing classes, inheritance, polymorphism and exception handling. This course covers GUI programming using swing.

Course Objectives:

- Understand object oriented programming
- Use of conditional statements and looping statements to solve problems associated with decision making and repetitions
- Write programs using more advanced JAVA features such as composition of objects, operator overloading, dynamic memory allocation, inheritance and polymorphism, file I/O, and exception handling
- Improve the problem solving skills
- Get knowledge about the basic concept of writing a program

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the basic concepts of java programming like control statements ,arrays and functions
CO2	Obtain knowledge about object oriented programming concepts
CO3	Acquire knowledge about file handling.
CO4	Apply Exception handling techniques and design GUI based applications using swing

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓		✓	✓
CO2	✓	✓			✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Review of Programming Concepts: program, identifiers, variables, constants, primitive data types, expressions, control statements, structured data types, arrays, functions.

Module 2: Object Oriented Concepts: Abstraction, encapsulation, objects, classes, methods, constructors, inheritance, polymorphism, static and dynamic binding, overloading, Abstract classes, Interfaces and Packages.

Module 3: File Handling: Byte Stream, Character Stream, File I/O Basics, File Operations, Serialization.

Module 4: Exception handling: Throw and Exception, Throw, try and catch Blocks, Multiple Catch Blocks, Finally Clause, Throwable Class, Types of Exceptions, java.lang Exceptions, Built-In Exceptions.GUI Design: GUI based I/O, Input and Message Dialog boxes, Swing components, Displaying text and images in windows.

Core Compulsory Readings

1. James Gosling, Bill Joy, Guy L. Steele Jr, GiladBracha, Alex Buckley, The Java Language Specification, Java SE 7 Edition, Addison-Wesley, 2013
2. Cay S. Horstmann, Core Java - Vol. I – Fundamentals, 10th Edition, Pearson, 2017
3. Deitel&Deitel, Java-How to Program (9th ed.), Pearson Education, 2012
4. Richard Johnson, An Introduction to Java Programming and Object-Oriented Application Development, Thomson Learning, 2006
5. Herbert Schildt, Java: The Complete Reference, 10th Edition, McGraw-Hill Education, 2018

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Create a class named ‘Student’ with string variable ‘name’ and integer variable ‘roll_no’. Assign the value of roll_no as 2 and name as John by creating an object of Student class.
2. Create a class to perform method overloading by changing the number of parameters
3. Create a class with a method to check if a number is less than 0 or not. If the number is less than 0, throw Arithmetic Exception, otherwise, print division is possible
4. Create a calculator using swing.
5. Create a simple program to read a text file MyFile.txt line by line using the Buffered Reader class.

Multi-Disciplinary Course

MSCSC03MDC04 MACHINE LEARNING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

The course is designed to offer an introduction to machine learning approach to Artificial Intelligence. The course provides an overview of the concepts of supervised and unsupervised learning methods. A discussion about deep learning is also included.

Course Objectives:

- To understand machine learning approach to Artificial Intelligence
- To understand fundamental issues and challenges of supervised and unsupervised learning techniques
- To design and implement supervised and unsupervised machine learning algorithms for real-world applications
- To appreciate the underlying mathematical relationships within and across Machine Learning algorithms

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand basic concepts of machine learning
CO2	Acquire knowledge on supervised learning and artificial neural networks.
CO3	Obtain knowledge on unsupervised learning.
CO4	Understand the concepts of deep learning architectures

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓	✓	✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓		✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction: Learning theory. Basics: Introduction to Machine Learning - Different Forms of Learning, Basics of Probability Theory, Linear Algebra and Optimization. Regression Analysis: Linear Regression, Ridge Regression, Lasso, Bayesian Regression, Regression with Basis Functions.

Module 2: Supervised learning: Linear separability and decision regions, Linear discriminants, Bayes optimal classifier, Linear regression, Standard and stochastic gradient descent, Lasso and Ridge Regression, Logistic regression, Support Vector Machines, Artificial Neural Networks ,Perceptron, Back propagation, , Decision Tree Induction, Overfitting, Pruning of decision trees, Bagging and Boosting, Dimensionality reduction and Feature selection. Support Vector Machines: Structural and empirical risk, Learning nonlinear hypothesis using kernel functions.

Module 3: Unsupervised learning: Clustering, Mixture models, Expectation Maximization, Spectral Clustering, Non-parametric density estimation. Dimensionality Reduction: Principal Component Analysis, Independent Component Analysis, Multidimensional Scaling, and Manifold Learning. Reinforcement Learning: Q-Learning, Temporal Difference Learning.

Module 4: Introduction to Deep Networks: Fundamentals of deep learning, Deep Feed forward Networks, Regularization for Deep Learning, Optimization for Training Deep Models, Introduction to Convolutional Networks, Sequence Modelling using Recurrent Nets, overview of LSTM, fundamentals of Generative adversarial Network.

Core Compulsory Readings

1. E. Alpaydin, Introduction to Machine Learning, 3rd Edition, Prentice Hall of India, 2014
2. T Hastie, R Tibshirani and J Friedman, The Elements of Statistical Learning Data Mining, Inference, and Prediction, 2nd Edition, Springer, 2009.
3. C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2010
4. R. O. Duda, P. E. Hart, and D.G. Stork, Pattern Classification, John Wiley and Sons, 2012

5. Simon O. Haykin, Neural Networks and Learning Machines, Pearson Education, 2016

6. Tom Mitchell, Machine Learning, McGraw Hill Education, First Edition

Core Suggested Readings

1. R.O. Duda, P.E. Hart and D.G. Stork, Pattern Classification, Wiley, Second Edition

2. Jiawei Han and MichellineKamber, Data Mining: Tools and Techniques, Morgan Kaufmann, 3rd Edition

3. Hastie, Tibshirani and Friedman, Elements of Statistical Learning, Springer.

4.

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Multi-Disciplinary Course

MSCSC03MDC05 FOUNDATIONS IN DATA SCIENCE

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course is designed to offer the fundamental aspects of data science.

Course Objectives:

- To demonstrate proficiency with statistical analysis of data
- To develop the ability to build and assess data-based models
- To execute statistical analyses and interpret outcomes
- To apply data science concepts and methods to solve problems in real-world contexts and will communicate these solutions effectively

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Understand the fundamentals of data science
CO2	Illustrate various methods for statistical data modelling
CO3	Illustrate various methods for predictive modelling
CO4	Illustrate descriptive modelling, association rule mining data mining

Mapping of COs to PSOs

	PSO1	**PSO2**	**PSO3**	**PSO4**	**PSO5**	**PSO6**
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction: Introduction data acquisition, data pre-processing techniques including data cleaning, selection, integration, transformation and reduction, data mining, interpretation.

Module 2: Statistical data modelling: Review of basic probability theory and distributions, correlation coefficient, linear regression, statistical inference, exploratory data analysis and visualization.

Module 3: Predictive modelling: Introduction to predictive modelling , decision tree, nearest neighbour classifier and naïve Baye's classifier, classification performance evaluation and model selection.

Module 4: Descriptive Modeling: Introduction to clustering, partitional, hierarchical, and density-based clustering (k-means, agglomerative, and DBSCAN), outlier detection, clustering performance evaluation. Association Rule Mining: Introduction to frequent pattern mining and association rule mining, Apriori algorithm, measures for evaluating the association patterns. Text Mining: Introduction of the vector space model for document representation, term frequency-inverse document frequency (tf-idf) approach for term weighting, proximity measures for document comparison, document clustering and text classification.

Core Compulsory Readings

1. W. McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy and iPython, 2nd Ed., O'Reilly, 2017.

Core Suggested Readings

1. P. Tan, M. Steinbach, AKarpatne, and V. Kumar, Introduction to Data Mining, 2nd Ed., Pearson Education, 2018.
2. G James, D Witten, T Hastie and R Tibshirani, An Introduction to Statistical Learning with Applications in R, Springer Texts in Statistics, Springer, 2013.
3. G. Grolemund, H. Wickham, R for Data Science, 1st Ed., O'Reilly, 2017.

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	Percentage
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Multi-Disciplinary Course

MSCSC03MDC06 DIGITAL SIGNAL PROCESSING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course offers the fundamentals of digital signal processing. Topics such as modern digital signal processing algorithms and applications, analysis of discrete time signals are included.

Course Objectives:

- To study the modern digital signal processing algorithms and applications
- To study the analysis of discrete time signals
- To achieve comprehensive knowledge to use of digital systems in real time applications
- To apply the algorithms for a wide area of recent applications.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the fundamentals of digital signal processing.
CO2	Understand various classes of digital signals.
CO3	Illustrate the time domain representations of signals and systems.
CO4	Explain the frequency analysis of signals.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3		✓		✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Signals and Signal Processing - Characterization and classification of Signals, Typical signal processing operations, Typical Signal Processing Applications, Advantage of Digital Signal Processing.

Module 2: Classification of signals –Introduction to vector space - The concept of frequency in continuous and discrete time signals -Sampling of analog signals – Sampling theorem – Quantization and Coding – Digital to analog conversion.

Module 3: Time Domain Representation of signals and systems - Discrete time signals, Operations on sequences, Discrete time Systems, Linear Time invariant Discrete Time Systems - convolution sum – correlation of discrete time signals, Z-Transform.

Module 4: Frequency Analysis of Signals - Frequency Analysis of Continuous Time Signals, Frequency Analysis of Discrete Time Signals, Fourier Transform of discrete time signals –Discrete Fourier Transform (DFT). FFT (Qualitative idea only) - Wavelet Transform - FIR and IIR Filters.

Core Compulsory Readings

1. Proakis, John G. and Dimitris G. Manolakis. Digital signal processing: principles, algorithms and applications. Pearson Education India, 2001

Core Suggested Readings

1. Roberts, Michael J. Signals and systems: analysis using transform methods and MATLAB. McGraw-Hill Higher Education, 2011
2. Oppenheim, Alan V., and Ronald W. Schafer. Digital Signal Processing [by] Alan V. Oppenheim [and] Ronald W. Schafer. Prentice-Hall, 1975
3. Antoniou, Andreas. Digital signal processing. McGraw-Hill, 2016
4. Rainer, Lawrence R., Bernard Gold, and C. K. Yuen. Theory and application of digital signal processing. Prentice-Hall, 2007

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Multi-Disciplinary Course

MSCSC03MDC07 QUANTUM COMPUTING AND INFORMATION THEORY

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course offers the basics of quantum computing information theory.

Course Objectives:

- To study the basics of complex vector spaces
- To study the fundamental concepts of quantum mechanics as applied in quantum computing.
- To learn the architecture and algorithms in quantum computing
- To study the fundamentals of quantum computations

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the fundamentals of quantum computing
CO2	Understand the fundamentals concepts of mathematics and physics required to learn quantum computing
CO3	Illustrate the principles of quantum circuits
CO4	Explain the various algorithms and applications in quantum computing

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2		✓		✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓		✓	✓	

COURSE CONTENTS

Module 1: Introduction to Quantum Computation: Quantum bits, Bloch sphere representation of a cubit, multiple cubits.

Module 2: Background Mathematics and Physics: Hilbert space, Probabilities and measurements, entanglement, density operators and correlation, basics of quantum mechanics, Measurements in bases other than computational basis.

Module 3: Quantum Circuits: single qubit gates, multiple qubit gates, design of quantum circuits.

Module 4: Quantum Information and Cryptography: Comparison between classical and quantum information theory. Bell states.Quantum teleportation.Quantum Cryptography, no cloning theorem. Quantum Algorithms: Classical computation on quantum computers. Relationship between quantum and classical complexity classes.Deutsch's algorithm, Deutsch's-Jozsa algorithm, Shor factorization, Grover search.

Core Compulsory Readings

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press.-2002.
2. Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II: Basic Tools and Special Topics, World Scientific.-2004.
3. Pittenger A. O., An Introduction to Quantum Computing Algorithms-2000.

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Notes

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

End of S3 Multi-Disciplinary Courses

END OF SEMESTER III

Semester IV

ELECTIVE COURSE – POOL C

MSCSC04DSE10 SOFTWARE ENGINEERING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: This course provides an overview about Software Engineering. It provides the fundamental aspects of the topic and covers the various process models available for creating software projects. It provides insight to the requirement engineering and discusses various techniques for modeling requirements. Fundamentals of software design and component level design in particular is also covered. The course also provides insight into the principles of user interface design, project estimation, scheduling and testing strategies.

Course Objectives:

- To introduce the basic concepts of software engineering
- To introduce various software process models
- To build an understanding on various processes of software development

Course Outcome:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Identify the apt process model to build a software project
CO2	Illustrate the gathering of requirements and translating them into an appropriate model
CO3	Illustrate the principles of various aspects of software design
CO4	Identify the testing strategies suitable for a project and illustrate various methods for project estimation

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓		✓	✓	✓	✓
CO2		✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Fundamentals of Software Engineering: Evolution, Software Crisis, Definition, Layered Technology, Software Process, Framework Activities, Kinds of Process Flows. Process Models: Waterfall Model, V Model, Incremental Process Model, Evolutionary Process Models, Concurrent Model, Concepts of Agile Development, Agile Process Models: Extreme Programming (XP), Adaptive Software Development, Dynamic system Development Method(DSDM), Scrum

Module 2: Requirements Engineering: Tasks, Requirements Elicitation: Collaborative Requirement Gathering, QFD, Usage Scenarios. UML Fundamentals. Developing Use Cases, Developing Use Case Diagrams. Requirements Modeling: Elements, Scenario based Modelling, Data Modeling, Developing ER Diagrams, Fundamentals of DFD, Flow Oriented Modeling

Module 3: Design within the Context of Software Engineering, Design Concepts: Abstraction, Architecture, Patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Refactoring. Design Model. Component-Level Design: Software Component, Different Views of a Component, Designing Class- Based Components. User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface analysis, Interface design steps

Module 4: Software Project Estimation: Estimation based on LOC, FP, Process and Use Cases, COCOMO II Model, Estimation for Web Applications. Scheduling: Time-line Charts (GANTT Charts), Scheduling for Web App Projects. Software Testing: Unit Testing, Integration Testing, Regression Testing, Smoke Testing.

Core Compulsory Readings

Roger S Pressman, Software Engineering: A Practitioner’s Approach, 7th Edition, McGraw-Hill International Edition, 2010

Core Suggested Readings

1. Richard Fairey, Software Engineering concepts, Tata McGraw-Hill 2009 reprint
2. Ian Sommerville, Software Engineering, 6th Ed., Addison Wesley
3. Waman S Jawadekar, Software Engineering Principles and Practice, Tata McGraw Hill, 2004

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	Percentage
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Identity the framework activities of a software process

2. Identity the elements UML diagram

3. State the golden rules of interface design

4. State the goals of software project estimation

5. Differentiate between incremental process model and evolutionary process model

6. Prepare the use case for a sample project work (Assume a suitable project work and a sample event involving an actor)

7. Prepare short note on any two design concepts

8. Illustrate LOC based estimation with a suitable example

9. Explain various process models

10. Explain the creation of flow oriented model with a suitable example

11. Explain the creation of data model model with a suitable example

12. Illustrate any one golden rules of interface design

ELECTIVE COURSE – POOL C

MSCSC04DSE11 CYBER PHYSICAL SYSTEMS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: A cyber-physical system (CPS) is an interconnected network of physical objects, devices, and systems that communicate and interact with each other through computational means. It involves the integration of digital and physical components to monitor, control, and optimize processes in various domains, including transportation, healthcare, and manufacturing. CPSs enable the exchange of data and information between the physical and virtual worlds, leading to improved efficiency, automation, and decision-making. However, they also pose challenges related to security, privacy, and the reliability of complex software and hardware systems.

Course Objectives:

- Understand the basic concepts and principles of CPS.
- Gain knowledge of the integration of physical and computational components in CPS.
- Learn to design and analyse CPS applications effectively.
- Develop skills in modelling and simulating CPS for predicting system behaviour.
- Address security and privacy challenges in CPS to ensure data and system integrity.

Course Outcomes:

SL #	Course Outcomes
CO1	Understand Cyber Physical systems and its various components
CO2	Analyse the stability and performance level analysis of CPS
CO3	Applications of state machines for CPS design
CO4	Analyse and verify the correctness of CPS implementations against system requirements and security.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Cyber-Physical Systems (CPS) in the real world, Basic principles of design and validation of CPS, CPS HW platforms: Processors, Sensors, Actuators, CPS Network, CPS SW stack RTOS, Scheduling Real Time control tasks. Principles of Automated Control Design: Dynamical Systems and Stability, Controller Design Techniques.

Module 2: Stability Analysis: CLFs, MLFs, stability under slow switching, Performance under Packet drop and Noise. CPS : From features to software components, Mapping software components to ECUs, CPS Performance Analysis : effect of scheduling, bus latency, sense and actuation faults on control performance, network congestion, Formal Methods for Safety Assurance of Cyber-Physical Systems.

Module 3: Advanced Automata based modelling and analysis: Basic introduction and examples, Timed and Hybrid Automata, Definition of trajectories, zenoness, Formal Analysis: Flow pipe construction, reachability analysis, Analysis of CPS Software, Weakest Pre-conditions, Bounded Model checking.

Module 4: Hybrid Automata Modelling :Flowpipe construction using Flowstar, SpaceX and Phaver tools, CPS SW Verification: Frama-C, CBMC, Secure Deployment of CPS : Attack models, Secure Task mapping and Partitioning, State estimation for attack detection, Automotive Case study : Vehicle ABS hacking, Power Distribution Case study : Attacks on Smart grid.

Core Compulsory Readings

1. E. A. Lee and S. A. Seshia, “Introduction to Embedded Systems: A Cyber-Physical Systems Approach”, 2011.

2. R. Alur, “Principles of Cyber-Physical Systems,” MIT Press, 2015.

Core Suggested Readings

1. T. D. Lewis “Network Science: Theory and Applications”, Wiley, 2009.
2. P. Tabuada, “Verification and control of hybrid systems: a symbolic approach”, Springer-Verlag 2009.
3. C. Cassandras, S. Lafortune, “Introduction to Discrete Event Systems”, Springer 2007.
4. Constance Heitmeyer and Dino Mandrioli, “Formal methods for real-time computing”, Wiley publisher, 1996

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is the definition of a cyber-physical system (CPS) and how does it differ from traditional systems?
2. What are the key components involved in a typical cyber-physical system?
3. How does the integration of physical and computational components in CPS enhance system functionality?

4. What are some real-world applications of cyber-physical systems in different industries?
5. What are the challenges and considerations in ensuring the security and privacy of cyber-physical systems?
6. What are the techniques and methodologies used for modeling and simulating cyber-physical systems?
7. How does real-time control play a crucial role in the operation of cyber-physical systems?
8. What are the implications of scalability and interoperability in the design and implementation of cyber-physical systems?
9. How do cyber-physical systems contribute to improving efficiency, automation, and decision-making in various domains?
10. What are the emerging trends and future directions in the field of cyber-physical systems?

ELECTIVE COURSE – POOL C

MSCSC04DSE12 DIGITAL FORENSICS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: This course provides an in-depth exploration of digital forensics, focusing on the investigation and analysis of digital evidence in criminal and civil cases. Participants will learn essential techniques for acquiring, preserving, and analyzing digital data from various devices, including computers, mobile devices, and networks. Through hands-on exercises and real-world case studies, students will develop expertise in extracting and interpreting digital artifacts, recovering deleted information, and presenting findings in a legally admissible manner. By the end of the course, students will possess the necessary skills to conduct digital investigations and contribute to the field of digital forensics.

Course Objectives:

- Gain a comprehensive understanding of the principles and methodologies used in digital forensic investigations.
- Develop proficiency in acquiring and preserving digital evidence from a variety of devices and sources.
- Learn how to analyze and interpret digital artifacts to reconstruct events and timelines accurately.
- Familiarize oneself with the tools and techniques used to recover deleted or hidden information from digital storage media.
- Acquire the knowledge and skills required to present digital evidence effectively in legal proceedings, adhering to the guidelines and standards of forensic practices.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the Computer forensics profession and its various artifacts.
CO2	Understand the technical requirements for use of data acquisition.
CO3	Computer Forensics Analysis and Validation for digital data including email, mobile based systems.
CO4	Understand various Digital Forensics Real time applications

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1				✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Computer Forensics and Investigation: Understanding computer forensics, Preparing for Computer Investigations, Corporate High Tech Investigation. Data Acquisition and Recovery. Storage formats, Using acquisition tools, Data Recovery: RAID Data acquisition.

Module 2: Processing Crime and Incident Scene: Identifying and collecting evidence, Preparation for search, Seizing and Storing Digital evidence. Computer Forensics tools (Encase) and Windows Operating System. Understanding file structure and file system, NTFS disks, Disk Encryption and Registry. Manipulation. Computer Forensics software and hardware tools.

Module 3: Computer Forensics Analysis and Validation: Data collection and analysis, validation of forensics data, addressing – data hiding technique. Email Investigation and Mobile device Forensics-Investigation e-mail crimes and Violations, Using specialized E-mail forensics tools. Understanding mobile device forensics and Acquisition procedures.

Module 4: Role of Digital Forensics in Real time applications - SANS SIFT Investigative tool, PRO Discover Basic, Voltality, Sleuth Kit, CAINE investigative environment. Industry Trends.

Core Compulsory Readings

1. Bill Nelson, Amelia Philips, Christopher Steuart, Guide to Computer Forensics and Investigations, Fourth Edition, Cengage Learning, 2016.
2. David Lilburn Watson, Andrew Jones, Digital Forensics Processing and Procedures, Syngress, 2013.

Core Suggested Readings

1. Cory Altheide, Harlan Carvey, Digital Forensics with Open Source Tools, British Library Cataloguing-in-Publication Data, 2011
2. Greg Gogolin,Digital Forensics Explained,CRC Press, 2013.

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is digital forensics, and what is its role in modern investigations?

2. What are the key steps involved in conducting a digital forensic investigation?
3. What are the various types of digital evidence that can be collected and analyzed in a forensic investigation?
4. How can digital evidence be acquired and preserved to maintain its integrity and admissibility in court?
5. What are the common challenges and considerations when dealing with mobile device forensics?
6. How can deleted or hidden data be recovered and analyzed in digital forensics?
7. What are the different techniques used to analyze network traffic and identify potential cyber threats?
8. What are some legal and ethical considerations in digital forensics, particularly in terms of privacy and chain of custody?
9. How can digital forensics be applied in the investigation of cybercrimes, such as hacking or online fraud?
10. What are the emerging trends and advancements in digital forensics, and how do they impact the field?

ELECTIVE COURSE – POOL C

MSCSC04DSE13 BLOCK CHAIN AND CRYPTO CURRENCY TECHNIQUES

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: This course offers a comprehensive exploration of block chain technology and its applications in the realm of cryptocurrency. Students will delve into the fundamental concepts of blockchain, including distributed ledger technology, cryptographic techniques, and consensus mechanisms. Through hands-on projects and case studies, participants will gain practical experience in building blockchain-based systems and understanding the intricacies of cryptocurrencies such as Bitcoin and Ethereum. By the end of the course, students will have a solid understanding of blockchain's potential, its impact on industries, and the security considerations surrounding crypto currencies.

Course Objectives:

- Understand the fundamental principles of block chain technology, including distributed ledger, cryptographic techniques, and consensus mechanisms.
- Gain practical knowledge of building and deploying blockchain-based systems for various applications.
- Explore the functionalities and underlying mechanisms of popular cryptocurrencies such as Bitcoin and Ethereum.
- Develop skills in evaluating the potential impact of blockchain technology across industries and identifying suitable use cases.
- Acquire a comprehensive understanding of the security considerations and challenges associated with cryptocurrencies and blockchain implementations.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand and apply the fundamentals of Cryptography in Crypto currency
CO2	To gain knowledge about block chain and its various technological artifacts
CO3	Understand Bitcoin transactions
CO4	To educate the principles, practices and policies associated Bitcoin business

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2		✓		✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Cryptography and Cryptocurrencies: Foundations of Cryptography and security: Ciphers and secret messages, security attacks and services. Mathematical tools for cryptography: substitution techniques, modular arithmetic, Euclid's algorithm, finite fields, polynomial arithmetic. Design Principles of Block Ciphers: Theory of Block Cipher Design, Feistel cipher network structure, DES and Triple DES, modes of operation (ECB, CBC, OFB, CFB), strength of DES.

Module 2: Blockchain Achieves: Decentralization-Centralization vs. Decentralization-Distributed consensus, Consensus with- out identity using a blockchain, Incentives and proof of work. Simple Local Storage, Hot and Cold Storage, Splitting and Sharing Keys, Online Wallets and Exchanges, Payment Services, Transaction Fees, Currency Exchange Markets.

Module 3: Mechanics of Bitcoin: Bitcoin transactions, Bitcoin Scripts, Applications of Bitcoin scripts, Bitcoin blocks, The Bit- coin network, Limitations and improvements. Bitcoin Mining :The task of Bitcoin miners, Mining Hardware, Energy consumption and ecology, Mining pools, Mining incentives and strategies. Bitcoin and Anonymity: Anonymity Basics, How to De-anonymize Bitcoin, Mixing, Decentralized Mixing, Zerocoin and Zerocash

Module 4: Community, Politics, and Regulation: Consensus in Bitcoin, Bitcoin Core Software, Stakeholders: Who‟s in Charge, Roots of Bitcoin, Governments Notice on Bitcoin, Anti Money Laundering Regulation, New York‟s Bit License Proposal. Bitcoin as a Platform: Bitcoin as an Append only Log, Bitcoins as Smart Property, Secure Multi Party Lotteries in Bitcoin, Bitcoin as Public Randomness, Source-Prediction, Markets, and Real World Data Feeds.

Core Compulsory Readings

1. Narayanan, A., Bonneau, J., Felten, E., Miller, A., and Goldfeder, S. (2016). Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press.

2. Antonopoulos, A. M. (2014). Mastering Bitcoin: unlocking digital cryptocurrencies. OReilly Media, Inc.".

Core Suggested Readings

1. William Stallings, Cryptography and Network Security, Pearson 2004.

2. Franco, P. (2014). Understanding Bitcoin: Cryptography, engineering and economics. John Wiley and Sons.

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is the concept of distributed ledger and how does it contribute to the security and transparency of blockchain technology?
2. How are cryptographic techniques used in blockchain to ensure data integrity and privacy?
3. What are the different consensus mechanisms employed in blockchain networks, and what are their advantages and disadvantages?
4. How does the Bitcoin blockchain work, and what role does it play in enabling peer-to-peer transactions and decentralized digital currency?
5. What are smart contracts and how do they enhance the functionality of blockchain platforms like Ethereum?
6. What are the potential use cases for blockchain technology beyond cryptocurrency, and how can it revolutionize industries such as supply chain, healthcare, or finance?
7. What are the major security vulnerabilities and risks associated with cryptocurrencies, and how can they be mitigated?
8. What is the process of mining in blockchain networks, and how does it contribute to the validation and security of transactions?
9. How can blockchain be integrated with existing systems and technologies, and what challenges may arise in the process?
10. What are the regulatory and legal considerations surrounding cryptocurrencies and blockchain implementations, and how do they vary across different jurisdictions?

ELECTIVE COURSE – POOL C

MSCSC04DSE14 GRID AND CLOUD COMPUTING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course is designed to provide the fundamental concepts of natural language processing (using Python. Topics such as text processing of raw text, text categorization, information extraction from text and analysis of sentence structure are included in the discussion. Students are also exposed to developing solutions using Python for various operations related to NLP.

Course Objectives:

- To understand the fundamental concepts of Distributed Computing, Grid Computing and its large scale scientific applications
- To acquire knowledge on the concept of virtualization
- To gain knowledge about different cloud deployment models and cloud computing categories
- To learn implementations of open source grid middleware packages and Hadoop Framework
- To understand the security issues in the grid and the cloud environment

Course Outcome

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Awareness about grid and cloud computing.
CO2	Familiar with cloud deployment model.
CO3	Acquire the knowledge about open source grid middleware packages and Hadoop framework.
CO4	Awareness about grid and cloud security.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Evolution of Distributed computing: Scalable computing over the Internet – Technologies for network based systems – clusters of cooperative computers – Grid computing Infrastructures – cloud computing – service oriented architecture – Introduction to Grid Architecture and standards – Elements of Grid – Overview of Grid Architecture.Introduction to Open Grid Services Architecture (OGSA) – Motivation – Functionality Requirements – Practical & Detailed view of OGSA/OGSI – Data intensive grid service models – OGSA services.

Module 2: Cloud deployment models: public, private, hybrid, community – Categories of cloud computing: Everything as a service: Infrastructure, platform, software, Security as a Service – Pros and Cons of cloud computing – Implementation levels of virtualization – virtualization structure – virtualization of CPU, Memory and I/O devices, Desktop Virtualization – virtual clusters and Resource Management – Virtualization for data center automation. Tools and Products available for Virtualization.

Module 3: Open source grid middleware packages – Globus Toolkit (GT4) Architecture , Configuration – Usage of Globus – Main components and Programming model – Introduction to Hadoop Framework – Mapreduce, Input splitting, map and reduce functions, specifying input and output parameters, configuring and running a job – Design of Hadoop file system, HDFS concepts, command line and java interface, dataflow of File read & File write.

Module 4: Trust models for Grid security environment – Authentication and Authorization methods – Grid security infrastructure – Cloud Infrastructure security: network, host and application level – aspects of data security, provider data and its security, Identity and access management architecture, IAM practices in the cloud, SaaS, PaaS, IaaS availability in the cloud, Key privacy issues in the cloud.

Core Compulsory Readings

1. Kai Hwang, Geoffery C. Fox and Jack J. Dongarra, Distributed and Cloud Computing: Clusters, Grids, Clouds and the Future of Internet, First Edition, Morgan Kaufman Publisher, an Imprint of Elsevier, 2012

Core Suggested Readings

1. Jason Venner, Pro Hadoop - Build Scalable, Distributed Applications in the Cloud, A Press, 2009
2. Tom White, Hadoop The Definitive Guide, First Edition. O Reilly, 2009
3. Bart Jacob (Editor), Introduction to Grid Computing, IBM Red Books, Vervante, 2005
4. Ian Foster, Carl Kesselman, The Grid: Blueprint for a New Computing Infrastructure, 2nd Edition, Morgan Kaufmann
5. Frederic Magoules and Jie Pan, Introduction to Grid Computing, CRC Press, 2009
6. Daniel Minoli, A Networking Approach to Grid Computing, John Wiley Publication, 2005
7. Barry Wilkinson, Grid Computing: Techniques and Applications, Chapman and Hall, CRC, Taylor and Francis Group, 2010
8. Kris Jamsa, Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and more, Jones & Bartlett Learning Company, 2013
9. R. BUYYA, C. VECCHIOLA, S T. SELVI, Mastering Cloud Computing, Mc Graw Hill (India) Pvt Ltd., 2013

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is the fundamental difference between Grid Computing and Cloud Computing, and how do they approach resource sharing differently?

2. How does Grid Computing utilize distributed resources from multiple locations to perform complex computations or solve large-scale problems?

3. Describe the key characteristics of Cloud Computing and its various service models, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

4. How does virtualization play a crucial role in both Grid and Cloud Computing environments, and what are its benefits?

5. What are the major challenges in managing security and privacy in Grid and Cloud Computing, and how are they addressed in practice?

6. Explain the concept of elasticity in Cloud Computing and how it enables on-demand resource allocation and scalability.

7. What is the role of load balancing in Grid and Cloud Computing, and how does it contribute to optimizing resource utilization and performance?

8. Discuss the differences between centralized and decentralized resource management approaches in Grid and Cloud Computing.

9. How are fault tolerance and high availability achieved in both Grid and Cloud Computing infrastructures?

10. What are the potential environmental impacts of Grid and Cloud Computing, and how can sustainable practices be adopted in these computing paradigms?

ELECTIVE COURSE – POOL C

MSCSC04DSE15 EMBEDDED SYSTEM

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: This course provides a comprehensive introduction to embedded systems, focusing on the design, development, and implementation of real-time computing systems. Students will gain hands-on experience in programming microcontrollers, understanding hardware-software interfaces, and designing efficient algorithms for resource-constrained devices. Through practical projects, participants will learn to develop embedded systems for various applications such as automotive, medical devices, and Internet of Things (IoT) devices. By the end of the course, students will have a solid foundation in embedded systems and be equipped with the skills to tackle complex challenges in this rapidly evolving field.

Course Objectives:

- Understand the fundamental concepts of embedded systems, including hardware-software interactions and real-time computing.
- Gain proficiency in programming microcontrollers and designing efficient algorithms for resource-constrained devices.
- Develop hands-on skills in building and testing embedded systems for diverse applications such as automotive, medical devices, and IoT devices.
- Learn to interface sensors, actuators, and other peripherals with microcontrollers to create functional embedded systems.
- Acquire problem-solving abilities to tackle challenges in the design, development, and debugging of embedded systems.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire basic knowledge on embedded system.
CO2	Understand different serial and parallel communication standards and devices.
CO3	Understand various functions of real-time operating system.
CO4	Understand various memory management schemes and programming concepts of embedded system.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓			✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Embedded Systems– Components of embedded system hardware–Software embedded into the system – Embedded Processors - CPU architecture of ARM processor (ARM9) – CPU Bus Organization and Protocol. Design and Development life cycle model - Embedded system design process – Challenges in Embedded system design.

Module 2: Serial Communication Standards and Devices - UART, HDLC, SCI and SPI. Serial Bus Protocols - I2C Bus, CAN Bus and USB Bus. Parallel communication standards ISA, PCI and PCI-X Bus.

Module 3: Real-time operating systems - Services- Goals – Structures -Kernel - Process Management – Memory Management –Device Management – File System Organization. Micro C/OS-II RTOS - System Level Functions – Task Service Functions – Memory Allocation Related Functions Semaphore Related Functions. Study of other popular Real-time Operating Systems.

Module 4: Memory devices and systems - memory map – DMA - I/O Devices – Interrupts - ISR – Device drivers for handling ISR – Memory Device Drivers – Device Drivers for on-board bus. Programming concepts of embedded programming – Features of Embedded C++ and Embedded Java (basics only). Software Implementation, Testing, Validation and debugging, system-on-chip.Design Examples: Mobile phones, ATM machine, Set top box.

Core Compulsory Readings

1. David E. Simon, An Embedded Software Primer, Pearson Education Asia, First Indian Reprint 2000.
2. Wayne Wolf, Computers as Components: Principles of Embedded Computing System Design, Morgan Kaufman Publishers - Elsevier 3ed, 2008.

Core Suggested Readings

1. Frank Vahid and Tony Givargis, Embedded Systems Design – A Unified Hardware / Software Introduction, John Wiley, 2002.
3. Iyer - Embedded Real time Systems, 1e, McGraw Hill Education New Delhi, 2003.
4. K.V. Shibu, Introduction to Embedded Systems, 2e, McGraw Hill Education India, 2016.
5. Lyla B. Das, Embedded Systems: An Integrated Approach, 1/e ,Lyla B. Das, Embedded Systems, 2012.
6. Rajkamal,Embedded Systems Architecture, Programming and Design, TMH, 2003.
7. Steve Heath,Embedded Systems Design, Newnes – Elsevier 2ed, 2002.
8. Tammy Noergaard, Embedded Systems Architecture, A Comprehensive Guide for Engineers and Programmers, Newnes – Elsevier 2ed, 2012.

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is an embedded system and how does it differ from general-purpose computing systems?
2. What are the key components of an embedded system and how do they interact with each other?
3. How can real-time computing be achieved in embedded systems? What are the challenges associated with it?
4. What are some common programming languages used in embedded system development, and what factors influence the choice of language?
5. How do microcontrollers differ from microprocessors in terms of their architecture and functionality?
6. What are the considerations for selecting sensors and actuators for an embedded system, and how are they interfaced with microcontrollers?
7. How can power consumption be optimized in embedded systems to extend battery life and enhance energy efficiency?
8. What are the security challenges and considerations in embedded systems, particularly in the context of IoT devices?
9. What are the key steps involved in the development lifecycle of an embedded system, from design to deployment?
10. How does debugging and testing differ in embedded systems compared to traditional software development, and what tools are commonly used for this purpose?

ELECTIVE COURSE – POOL C

MSCSC04DSE16 HIGH PERFOMANCE COMPUTING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course on High-Performance Computing (HPC) provides a comprehensive understanding of the principles, architectures, and techniques involved in executing computationally intensive tasks. Students will learn about parallel computing, distributed systems, and the utilization of HPC clusters for scientific simulations and data analysis. The course covers topics such as programming models, optimization strategies, and performance evaluation to maximize the efficiency of HPC applications. Through hands-on exercises and projects, students will gain practical experience in developing and implementing parallel algorithms, as well as utilizing advanced tools and frameworks for HPC. By the end of the course, students will possess the skills necessary to leverage HPC resources and harness the full potential of high-performance computing in scientific research and computational-driven fields.

Course Objectives:

- Understand the principles and foundations of high-performance computing and its importance in scientific and computational-driven fields.
- Gain knowledge of parallel computing architectures and techniques used in high-performance computing.
- Develop skills in programming and optimizing algorithms for efficient execution on HPC clusters.
- Learn to utilize advanced tools and frameworks for managing and analyzing large-scale data in high-performance computing environments.
- Apply the acquired knowledge and skills to solve complex computational problems and conduct scientific simulations using high-performance computing resources.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire knowledge on basics of parallel programming models.
CO2	Understand the memory organization and architectural features of HPC.
CO3	Obtain knowledge on synchronization and performance analysis of parallel algorithms.
CO4	Acquire knowledge on power aware processing techniques and quantum computers.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Levels of parallelism (instruction, transaction, task, thread, memory, function)- Models (SIMD, MIMD, SIMT, SPMD, Dataflow Models, Demand-driven Computation etc)- Architectures: N-wide superscalar architectures, multi-core, multi-threaded.

Module 2: Processor Architecture, Interconnect, Communication, Memory Organization, and Programming Models in high performance computing architectures: (Examples: IBM CELL BE, Nvidia Tesla GPU, Intel Larrabee Microarchitecture and Intel Nehalem microarchitecture- Memory hierarchy and transaction specific memory design- Thread Organization.

Module 3: Synchronization- Scheduling- Job Allocation-Job Partitioning- Dependency Analysis- Mapping Parallel Algorithms onto Parallel Architectures- Performance Analysis of Parallel Algorithms.

Module 4: Bandwidth Limitations- Latency Limitations- Latency Hiding/Tolerating Techniques and their limitations- Power-aware Processing Techniques-Power-aware Memory Design- Power-aware

Interconnect Design-Software Power Management. Petascale Computing-Optics in Parallel Computing- Quantum Computers- Recent developments in Nanotechnology and its impact on HPC.

Core Compulsory Readings

1. George S. Almasi and AlanGottlieb, Highly Parallel Computing, Benjamin Cumming Publishers.

2. Kai Hwang ,Advanced Computer Architecture: Parallelism, Scalability, Programmability, McGraw Hill 1993

Core Suggested Readings

1. David Culler, Jaswinder Pal Singh, Anoop Gupta, Parallel Computer Architecture: A hardware/Software Approach, Morgan Kaufmann, 1999.
2. K. Hwang& Z. Xu, Scalable Parallel Computing – Technology, Architecture, Programming., McGraw Hill 1998.
3. William James Dally and BrianTowles, Principles and Practices on Interconnection Networks, Morgan Kauffman 2004.
4. Hubert Nguyen , GPU Gems 3, Addison Wesley, 2008, (Chapter 29 to Chapter 41)
5. AnanthGrama, Anshul Gupta, George Karypis, and Vipin Kumar, Introduction to Parallel Computing, , 2nd edition, Pearson, 2003.
6. David A. Bader (Ed.), Petascale Computing: Algorithms and Applications, Chapman & Hall/CRC, 2008.

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is high-performance computing (HPC) and how does it differ from traditional computing?
2. What are the main characteristics and advantages of parallel computing in the context of HPC?
3. What are some common architectures used in HPC systems, and how do they contribute to improved performance?
4. How can HPC clusters be utilized for scientific simulations and data analysis?
5. What programming models and languages are commonly used in HPC applications?
6. How can algorithms be optimized for efficient execution on HPC systems?
7. What are the key performance evaluation metrics used to assess the efficiency of HPC applications?
8. What are some tools and frameworks available for managing and analyzing large-scale data in HPC environments?
9. How does HPC contribute to advancements in fields such as weather forecasting, computational biology, and artificial intelligence?
10. What are the current challenges and future trends in high-performance computing?

ELECTIVE COURSE – POOL C

MSCSC04DSE17 DATA AND INFORMATION VISUALIZATION

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course focuses on building creative and technical skills to transform data into visual reports for the purpose of a better understanding. Students will learn to organize and visualize data with an emphasis on applying design principles to produce clear, elegant graphs from the data.

Course Objectives:

- Employ best practices in data visualization to develop charts, maps, tables, and other visual representations of data.
- Implement visualizations in Python

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand different data visualization techniques such as scatterplot, treemap, histogram and gantt chart.
CO2	Acquire knowledge on design principles of information visualization.
CO3	Use information visualization methods such as cladogram, cartogram, heatmap and dendrogram.
CO4	Understand volume rendering techniques.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3	✓	✓		✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Data and types of data, Data variability, uncertainty and context. Basics of Data Visualization: Definition of Data Visualization, Need for Visualization, How to visualize data, General types of Data Visualization, pros and Cons of Data Visualization. Visualization Components: Visual cues, Coordinate systems, Scales, and Context. Diagrams used for data visualization: Bar chart, Histogram, Scatter plot, Scatter plot, Network, Streamgraph, Treemap, Gantt chart, Stripe graphic, Animated spiral graphic. Visualization based on types of data: Visualizing Categorical data , Visualizing Time series data , Visualizing Spatial data.

Module 2: Information Visualization: Definition, Objectives of Information Visualization. Visual representation of large scale collection of non- numerical information. Design Principles of Information Visualization: Principle of Simplicity, Principle of Proximity, Principle of Similarity, Principle of Closure, Principle of Connectedness, Principle of Good Continuation, Principle of Common fate, Principle of Familiarity, Principle of Symmetry.

Module 3: Methods for Information Visualization: Cartogram, Cladogram (phylogeny), Concept Mapping, Dendrogram (classification). Graph drawing, Heat map, Hyperbolic Tree, Tree mapping Multidimensional scaling. Information visualization reference model. Case study with real world problems.

Module 4: Scientific visualization: Introduction, Methods for visualizing two-dimensional and three dimensional data sets, volume visualization. Data Visualization using in Python matplotlib Module, pyplot, plot(), scatter, bar charts, Formatting, figure(), subplot(), text(), xlabel(), ylabel(), title(), Plotting Mathematical Functions.

Core Compulsory Readings

1. Nathan Yau, Data Points. Wiley Big Data Series

Core Suggested Readings

1. Healy, Kieran, Data Visualization: A Practical Introduction. Princeton University Press
2. Ben Bederson and Ben Shneiderman. The Craft of Information Visualization: Readings and Reflections. Morgan Kaufmann, 2003
3. Riccardo Mazza. Introduction to Information Visualization, Springer, 2009
4. Gowrishankar S, Veena A, "Introduction to Python Programming", 1st
5. Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
6. Alberto Fernandez Villan, Mastering OpenCV 4 with Python, Packt Publishing Ltd
7. Dr. R Nageswara Rao, Core Python Programming, 2nd edition, Dreamtech Publisher, 2019
8. Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 1st Edition,O'Reilly Media, 2017.
9. Wesley J. Chun, Core Python Programming, Second Edition, Publisher: Prent Hall Pub
10. Introduction to Computer Science using Python - Charles Dierbach, Wiley,2015

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Design a graph and describe the elements of data visualization with the help of it.
2. How do you explain positive and negative correlation in scatter plots.
3. Summarize design principles behind information visualization.
4. Design a cladogram and mark its parts.
5. Show the conversion from tree diagram to treemap with the help of an example diagram
6. Explain different scientific visualization techniques.
7. Discuss Any one surface rendering method.
8. Write a python program to plot y=x and $y=x^2$ as subplots.

ELECTIVE COURSE – POOL C

MSCSC04DSE18 INFORMATION RETRIEVAL SYSTEM

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course is to introduce fundamental concepts of information retrieval and information retrieval strategies. Emphasis is given to Distributed Information Retrieval and Information Retrieval algorithms. Students will learn about k-gram indexes for spelling correction.

Course Objectives:

- To understand the fundamentals of Information Retrieval.
- To apply machine learning techniques for text classification and clustering.
- To use information retrieval strategies.
- To learn different Information Retrieval algorithms.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Understand different information retrieval strategies.
CO2	Acquire knowledge on retrieval utilities.
CO3	Understand natural language systems to build semantic networks for text.
CO4	Obtain knowledge on user search techniques and system search techniques.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	
CO2	✓	✓		✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction: Retrieval strategies: vector space model, Probabilistic retrieval strategies: Simple term weights, Non binary independence model, Language models.

Module 2: Retrieval Utilities: Relevance feedback, clustering, N-grams, Regression analysis, Thesauri.

Module 3: Semantic networks, parsing Cross –Language: Information Retrieval: Introduction, Crossing the Language barrier.

Module 4: Integrating structured data and text. A historical progression, Information retrieval as relational application, Semi Structured search using a relational schema. Distributed Information Retrieval: A theoretical Model of Distributed retrieval, web search.

Core Compulsory Readings

1. David A. Grossman, Ophir Frieder, Information Retrieval – Algorithms and Heuristics, Springer, 2nd Edition (Distributed by Universal Press), 2004

Core Suggested Readings

1. Gerald J Kowalski, Mark T Maybury,"Information Storage and Retrieval Systems: Theory and Implementation", Springer, 2004.

2. SoumenChakrabarti, "Mining the Web: Discovering Knowledge from Hypertext Data", Morgan – Kaufmann Publishers, 2002.

3. Christopher D Manning, PrabhakarRaghavan, HinrichSchutze, "An Introduction to Information Retrieval" , Cambridge University Press, England, 2009.

TEACHING LEARNING STRATEGIES

- Lecturing

MODE OF TRANSACTION

- Lecture, Seminar, Discussion,Questioning and Answering

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Write a note on Taxonomy of Information retrieval Models.
2. Discuss in detail about User Relevance Feedback.
3. Explain in detail about components of the Information Retrieval System.
4. What are n-grams in the context of natural language processing?
5. What is cross-language information retrieval (CLIR) and why is it important?
6. What are the different types of clustering algorithms used in information retrieval?
7. What are some common regression models used in information retrieval?
8. What are the advantages and limitations of using thesauri in information retrieval systems?

ELECTIVE COURSE – POOL C

MSCSC04DSE19 BIOMETRIC IMAGE PROCESSING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course is designed to provide the fundamental concepts of Biometric Image Processing. Topics such as image enhancement, Hand and Iris Biometrics, and Morphological image processing are included in the discussion.

Course Objectives:

- To understand the fundamental concepts of Biometric Image Processing
- To acquire knowledge about Image enhancement
- To gain knowledge about Hand and Iris Biometrics
- To familiarize with Morphological image processing

Course Outcome:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the fundamentals of Image Processing
CO2	Illustrate the techniques for image enhancement
CO3	Illustrate the concepts of Morphological image processing
CO4	Describe various techniques for Hand and Iris Biometrics

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓		✓	✓	✓
CO2	✓	✓		✓	✓	✓
CO3		✓		✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Digital image representation, Fundamental steps in image processing, Elements of digital image processing system, Image sensing and acquisition, Sampling and quantization, Basic relationship between pixels, Transformation technology: Fourier transform - Discrete cosine transform.

Module 2: Image enhancement: Spatial domain methods: Basic gray level transformations - Histogram equalization - Smoothing spatial filter - Sharpening spatial filters - Laplacian, Frequency domain methods: Smoothing and sharpening filters – Ideal - Butterworth - Gaussian filters. Image Segmentation: Point- Line and edge detection - Thresholding - Global and multiple thresholding, Region splitting and merging.

Module 3: Morphological image processing: Fundamental concepts and operations, Dilation and Erosion, Compound operations, Morphological filtering, Basic morphological algorithms, Grayscale morphology. 2D and 3D face biometrics: Global face recognition techniques: Principal component analysis - Face recognition using PCA - Linear discriminant analysis - Face recognition using LDA, Local face recognition techniques: Geometric techniques - Elastic graph matching techniques, Hybrid face recognition techniques. 3D Face Image: Acquisition, Pre-processing and normalization, 3D face.

Module 4: Hand and Iris Biometrics: Characterization by minutiae extraction: Histogram equalization, Binarization, Skeletonization, Detection of minutiae, Matching, Performance evaluation, Preprocessing of iris images: Extraction of region of interest - Construction of noise mask – Normalization - Features extraction and encoding - Similarity measures between two iris codes. Fusion in biometrics: Multi-biometrics, Levels of fusion: Sensor level - Feature level - Rank level - Decision level fusion - Score level fusion.

Core Compulsory Readings

1. Rafael C Gonzalez, Richard E Woods and Steven L Eddins, Digital Image Processing, Pearson Education, New Delhi, 2013.

Core Suggested Readings

1. Amine Nait Ali and Regis Fournier, Signal and Image Processing for Biometrics, John Wiley and Sons, UK, 2012
2. Arun A Ross, Karthik Nandakumar and Jain A K, Handbook of Multibiometrics, Springer, New Delhi 2011
3. Oge Marques, Practical Image and Video Processing using MATLAB, John Wiley and Sons, New Jersey, 2011

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is the role of image processing in biometrics, and how does it enhance the accuracy and reliability of biometric systems?

2. How are fingerprint images processed and analyzed to extract unique minutiae points for biometric identification?

3. What are the key steps involved in facial recognition using biometric image processing techniques?

4. Explain the concept of iris recognition and the image processing methods used to capture and analyze iris patterns.

5. How does biometric image processing ensure robustness and security against spoof attacks or fraudulent attempts?

6. Discuss the challenges in processing and matching voice biometrics from speech signals and ways to overcome them.

7. How is vein pattern recognition achieved through image processing, and what advantages does it offer over other biometric modalities?

8. Explain the concept of gait analysis as a biometric modality and the image processing techniques employed in gait recognition systems.

9. How are palmprint images processed and feature-extracted for biometric authentication purposes?

10. What are the potential applications of biometric image processing beyond traditional security and identification scenarios?

ELECTIVE COURSE – POOL C

MSCSC04DSE20 JAVA PROGRAMMING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description:

This course introduces JAVA programming language with object-oriented programming principles. Emphasis is placed on developing skills on implementing classes, inheritance ,polymorphism and exception handling. This course covers GUI programming using swing.

Course Objective

- Understand object oriented programming.
- Use of conditional statements and looping statements to solve problems associated with decision making and repetitions.
- Write programs using more advanced JAVA features such as composition of objects, operator overloading, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling etc.
- Improve the problem solving skills.
- Get knowledge about the basic concept of writing a program.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the basic Java programming concepts.
CO2	Acquire the knowledge on OOP concepts.
CO3	Obtain knowledge on file handling in Java.
CO4	Use different exception handling techniques and design GUI based applications.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓			✓	✓
CO2	✓	✓	✓		✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Review of Programming Concepts: program, identifiers, variables, constants, primitive data types,expressions, control statements, structured data types, arrays, functions.

Module 2: Object Oriented Concepts: Abstraction, encapsulation, objects, classes, methods, constructors,inheritance, polymorphism, static and dynamic binding, overloading, Abstract classes, Interfaces and Packages.

Module 3: File Handling: Byte Stream, Character Stream, File I/O Basics, File Operations, Serialization.

Module 4: Exception handling: Throw and Exception, Throw, try and catch Blocks, Multiple Catch Blocks,Finally Clause, Throwable Class, Types of Exceptions, java.lang Exceptions, Built-In Exceptions.

GUI Design: GUI based I/O, Input and Message Dialog boxes, Swing components, Displaying text and images in windows.

Core Compulsory Readings

1. James Gosling, Bill Joy, Guy L. Steele Jr, GiladBracha, Alex Buckley, The Java Language
2. Specification, Java SE 7 Edition, Addison-Wesley, 2013.

Core Suggested Readings

1. Cay S. Horstmann, Core Java - Vol. I – Fundamentals, 10th Edition, Pearson, 2017.
2. Deitel&Deitel, Java-How to Program (9th ed.), Pearson Education, 2012.
3. Richard Johnson, An Introduction to Java Programming and Object-Oriented Application
4. Development, Thomson Learning, 2006.

5. Herbert Schildt, Java: The Complete Reference, 10th Edition, McGraw-Hill Education, 2018.

TEACHING LEARNING STRATEGIES

- Lecturing, Team Learning, Digital Learning

MODE OF TRANSACTION

- Lecture, Seminar, Discussion, Demonstration, Questioning and Answering, Audio, Video, Print

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. Create a class named 'Student' with string variable 'name' and integer variable 'roll_no'. Assign the value of roll_no as 2 and name as John by creating an object of Student class.
2. Create a class to perform method overloading by changing the number of parameters
3. Create a class with a method to check if a number is less than 0 or not. If the number is less than 0, throw Arithmetic Exception, otherwise, print division is possible
4. Create a calculator using swing.
5. Create a simple program to read a text file MyFile.txt line by line using the Buffered Reader class.

ELECTIVE COURSE – POOL C

MSCSC04DSE21 NATURE INSPIRED COMPUTING

Credit			**Teaching Hours**			**Assessment Weightage (%)**		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course provides an insight into various inspirations provided by nature and how these inspirations can be used to solve real life problems.

Course Objectives:

- To understand natural inspirations in problem solving
- To acquire knowledge about Ant Colony Optimization
- To gain knowledge about Swarm Intelligence
- To learn genetic algorithms
- To familiarize with DNA Computing

Course Outcome:

At the end of the Course, the Student will be able to:

SL #	**Course Outcomes**
CO1	Explain the inspirations provided by nature and illustrate how they can be used to solve real life problems.
CO2	Illustrate Ant Colony Optimization and Swarm Intelligence
CO3	Describe various aspects of genetic algorithms
CO4	Understand DNA Computing.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓		✓	✓	
CO2		✓		✓	✓	
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction: Nature Inspired Computing. Natural to Artificial Systems - Biological Inspirations in problem solving- Behavior of Social Insects: Foraging - Division of labor - Task Allocation – Cemetery Organization and Brood Sorting – Nest Building – Cooperative Transport.

Module 2: Ant Colony Optimization : Ant Behavior – Towards artificial Ants - Ant Colony Optimization – Problem solving using ACO – Extensions of Ant Systems – Applications. Swarm Intelligence: Introduction to Swarm Intelligence – Working of Swarm Intelligence – Particle Swarms Optimization– Applications.

Module 3: Introduction to Genetic Algorithms – Population Initialization – Choosing a Fitness Function – Selection – Crossover – Mutation – Reinsertion – Applications of Genetic Algorithms – Evolutionary Algorithms. Other Biological computing Methods – Immune System Algorithms – Cellular Automata – Linden Meyer Systems – Artificial Neural Networks – Simulated Annealing.

Module 4: Computing With New Natural Materials: DNA Computing: Motivation, DNA Molecule, Adelman’s experiment, Test tube programming language, Universal DNA Computers, PAM Model, Splicing Systems, Lipton's Solution to SAT Problem, Scope of DNA Computing, From Classical to DNA Computing.

Core Compulsory Readings

1. Stephen Olariu and Albert Y.Zomaya, Handbook of Bio - Inspired and Algorithms and Applications, Chapman and Hall, 2006
2. Marco Dorrigo, Thomas Stutzle, Ant Colony Optimization, PHI, 2004
3. Eric Bonabeau, Marco Dorrigo, Guy Theraulaz, Swarm Intelligence: From Natural to Artificial Systems, Oxford University Press, 2000
4. Mitchell, Melanie, Introduction to Genetic Algorithms, ISBN:0262133164, MIT Press,1996
5. Leandro Nunes de Castro, Fundamentals of Natural Computing, Basic Concepts, Algorithms

and Applications, Chapman & Hall/ CRC, Taylor and Francis Group, 2006

Core Suggested Readings

1. Floreano D. and Mattiussi C., Bio - Inspired Artificial Intelligence: Theories, Methods, and Technologies, MIT Press, Cambridge, MA, 2008
2. Leandro Nunes de Castro and Fernando .J, Recent Developments in Biological Inspired Computing, MIT Press, 200

TEACHING LEARNING STRATEGIES

- Lecturing, Demonstration

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes:

1. What is Nature-Inspired Computing, and how does it draw inspiration from natural systems and phenomena for problem-solving?

2. How does the Genetic Algorithm mimic the process of natural selection and evolution to optimize solutions in various domains?

3. Explain the concept of Swarm Intelligence and how it is utilized in algorithms like Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO).

4. What are the key characteristics of Artificial Neural Networks (ANNs) and how do they model the functioning of the human brain?

5. How does the behavior of social insects, such as bees and ants, influence the development of optimization techniques in Nature-Inspired Computing?

6. Describe the working principle of the Firefly Algorithm and how it optimizes solutions based on

the flashing patterns of fireflies.

7. What role does the concept of "survival of the fittest" play in Evolutionary Strategies (ES) and how are these strategies applied to complex problem spaces?

8. Discuss the advantages and disadvantages of using Nature-Inspired Computing techniques compared to traditional optimization algorithms.

9. How is Artificial Immune Systems (AIS) inspired by the human immune system, and what are the applications of AIS in problem-solving?

10. Explain how Nature-Inspired Computing is utilized in real-world applications, such as optimization in engineering, data analysis, and robotics.

ELECTIVE COURSE – POOL C

MSCSC04DSE22 BIG DATA ANALYTICS

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course "Big Data Analytics" equips students with the essential skills and knowledge needed to analyse and extract valuable insights from large and complex datasets. Students will learn about the challenges posed by big data, such as volume, velocity, variety, and veracity, and explore strategies to overcome these challenges effectively. Hands-on experience with popular big data technologies, including Apache Hadoop and Apache Spark, will enable students to process and analyse large-scale datasets using statistical and machine learning algorithms. By the end of the course, students will be prepared to tackle real-world big data challenges and make data-driven decisions to enhance their professional prospects.

Course Objectives:

- Understand the fundamentals of big data analytics.
- Develop proficiency in using big data technologies and tools.
- Apply statistical and machine learning techniques to analyse big data.
- Address challenges associated with big data processing and management.

Course Outcomes:

SL #	Course Outcomes
CO1	Acquire knowledge about Big Data Platforms and modern data analytic tools.
CO2	Information about Stream Data Model in Big Data.
CO3	Achieve the knowledge about Hadoop Distributed File System .
CO4	Acquire knowledge about applications of big data using pig and Hive.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓	✓			✓	
CO2		✓	✓	✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Big Data Platform – Challenges of Conventional Systems - Intelligent data analysis Nature of Data - Analytic Processes and Tools - Analysis vs Reporting - Modern Data Analytic Tools -Statistical Concepts: Sampling Distributions - Re-Sampling - Statistical Inference - Prediction Error.

Module 2: Introduction to Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window - Real time Analytics Platform (RTAP)Applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

Module 3: The Hadoop Distributed File System – Components of Hadoop- Analysing the Data with Hadoop- Scaling Out- Hadoop Streaming- Design of HDFS-Java interfaces to HDFS Basics-Developing a Map Reduce Application-How Map Reduce Works-Anatomy of a Map Reduce Job Run-Failures-Job Scheduling-Shuffle and Sort – Task execution - Map Reduce Types and Formats-Map Reduce Features. Setting up a Hadoop Cluster - Cluster specification - Cluster Setup and Installation – Hadoop Configuration-Security in Hadoop - Administering Hadoop – HDFS – Monitoring Maintenance- Hadoop benchmarks- Hadoop in the cloud.

Module 4: Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper - IBM InfoSphereBigInsights and Streams. Visualizations - Visual data analysis techniques, interaction techniques; Systems and applications

Core Compulsory Readings

1. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
2. Tom White, Hadoop: The Definitive Guide, 3rdEdn, O'reily Media, 2012.

Core Suggested Readings

1. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, Understanding BigData:Analytics for Enterprise Class Hadoop and Streaming Data, McGrawHill Pub, 2012
2. AnandRajaraman& Jeffrey D Ullman, Mining of Massive Datasets, Cambridge University Pres,2012.
3. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, John Wiley & sons, 2012.
4. Glen J. Myyat, Making Sense of Data, John Wiley & Sons, 2007
5. Pete Warden, Big Data Glossary, O'Reily, 2011.
6. Han, Kamber, Data Mining Concepts and Techniques, 3rdEdn, Morgan Kauffman, 2012.
7. Da Ruan, Guoquing Chen, Etienne E.Kere, Geert Wets, Intelligent Data Mining, Springer,2007
8. Paul Zikopoulos ,DirkdeRos , Krishnan Parasuraman , Thomas Deutsch , James Giles , DavidCorigan , Harness the Power of Big Data The IBM Big Data Platform, Tata McGraw Hill Pub,2012
9. Michael Mineli, Michele Chambers, AmbigaDhiraj, Big Data, BigAnalytics: Emerging BusinessIntelligence and Analytic Trends for Today's Business, WileyPublications,2013
10. Zikopoulos, Paul, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoopand Streaming Data, Tata McGraw Hill Pub, 2011

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What are the characteristics that define big data?

2. How does big data analytics contribute to informed decision-making?

3. What are the challenges associated with processing and analyzing big data?

4. Which technologies and tools are commonly used for big data processing?

5. How can Apache Hadoop and Apache Spark be leveraged for managing and analyzing large-scale datasets?

6. What are some strategies for cleaning and preprocessing big data before analysis?

7. How can statistical techniques be applied to extract meaningful insights from big data?

8. What are some machine learning algorithms commonly used in big data analytics?

9. How can data visualization techniques enhance the understanding and communication of insights derived from big data?

10. How can organizations effectively address the scalability and distributed computing requirements of big data processing?

ELECTIVE COURSE – POOL C

MSCSC04DSE23 ASTRONOMICAL IMAGE PROCESSING

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course on Astronomical Data Processing provides a comprehensive overview of the methods and techniques used to analyze and interpret data in the field of astronomy. Students will learn about the unique challenges posed by astronomical datasets and explore the specialized tools and algorithms employed in processing and extracting meaningful information from these vast datasets. The course covers topics such as data acquisition, calibration, image processing, spectral analysis, and statistical methods specific to astronomical data. Through hands-on projects and practical exercises, students will develop proficiency in utilizing software packages and programming languages commonly employed in astronomical data analysis. By the end of the course, students will have the skills to process, visualize, and interpret astronomical data, enabling them to contribute to scientific research and discovery in the field of astronomy.

Course Objectives:

- Understand the unique challenges associated with processing and analyzing astronomical data.
- Gain knowledge of the tools, algorithms, and techniques used in the processing of astronomical datasets.
- Develop proficiency in data acquisition, calibration, image processing, and spectral analysis specific to astronomy.
- Acquire practical skills in utilizing software packages and programming languages for astronomical data processing.
- Apply learned techniques to process, visualize, and interpret astronomical data for scientific research purposes.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Students will be able to effectively process and analyze astronomical data using specialized tools and algorithms.
CO2	Students will gain proficiency in acquiring, calibrating, and manipulating and visualizing astronomical datasets.
CO3	Acquire skills regarding the processing of astronomical image data.
CO4	Acquire skills in gathering spectral information by processing the astronomical data.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	✓
CO2	✓	✓	✓	✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to Astronomical Data Processing, Introduction to astronomical data and its characteristics, Overview of data acquisition techniques in astronomy, Introduction to observational astronomy and data collection methods, Introduction to astronomical databases and archives, Data reduction and calibration techniques in astronomy, Basic statistical concepts for astronomical data analysis

Module 2: Data Visualization and Exploration, Principles of data visualization for astronomical data, Techniques for exploring and visualizing large datasets, Introduction to plotting libraries and tools (e.g., Matplotlib, Plotly), Visualization of astronomical objects and phenomena (e.g., stars, galaxies, nebulae), Visualization of astronomical data in different formats (images, spectra, light curves), Data exploration techniques and interactive visualization tools

Module 3: Image Processing and Analysis, Image calibration and pre-processing techniques for astronomical images, Background subtraction and noise reduction methods, Object detection and extraction from astronomical images, Photometry techniques for measuring object brightness, Astrometry techniques for measuring object positions, Image stacking and co-addition for enhancing signal-to-noise ratio, Image registration and alignment techniques

Module 4: Spectral Data Processing and Analysis, Introduction to spectroscopy and spectral data in astronomy, Techniques for extracting and analysing spectral features, Spectral line identification and measurement, Spectral classification of astronomical objects, Redshift determination and its implications, Spectral analysis tools and software (e.g., IRAF, PyRAF, Specutils), Introduction to spectral modelling and interpretation

Core Suggested Readings

1. Richard Berry, Introduction to Astronomical Image Processing

2. Richard Berry, James Burnell, The Handbook of Astronomical Image Processing, 2005

3. J.-L. Starck , F. Murtagh , Astronomical Image and Data Analysis, 2006

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What are the primary sources of astronomical data and how are they collected?
2. How does data calibration contribute to accurate analysis and interpretation of astronomical data?
3. What are the common image processing techniques used in analyzing astronomical images?
4. How is spectral analysis utilized in understanding the composition and properties of celestial objects?
5. What statistical methods are commonly employed in analyzing astronomical data?
6. What software packages and programming languages are commonly used for processing astronomical data?
7. How do astronomers handle the challenges of working with large-scale datasets in terms of storage and computational resources?
8. What are the considerations for data visualization in presenting astronomical findings?
9. How can astronomical data processing contribute to the discovery of new celestial objects or phenomena?
10. What are the ethical considerations in handling and sharing astronomical data, particularly in the context of collaborative research efforts?

ELECTIVE COURSE – POOL C

MSCSC04DSE24 GEOGRAPHIC INFORMATION SYSTEM

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course on Geographical Information System (GIS) provides an introduction to the principles and applications of spatial data analysis. Students will learn how to capture, store, manipulate, analyze, and visualize geographic data. The course covers topics such as data collection techniques, spatial database management, geospatial analysis, and cartographic design. Through hands-on exercises and projects, students will develop practical skills in utilizing GIS software and tools for solving real-world problems in various domains, such as urban planning, environmental management, and transportation logistics.

Course Objectives:

- Understand the fundamental principles and concepts of Geographical Information Systems (GIS).
- Gain proficiency in using GIS software for data manipulation, analysis, and visualization.
- Develop skills in collecting, managing, and organizing spatial data within a GIS environment.
- Apply geospatial analysis techniques to solve practical problems in fields such as urban planning, environmental sciences, and logistics.
- Acquire knowledge of cartographic design principles to effectively communicate spatial information through maps and visualizations.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Acquire the basic knowledge of GIS.
CO2	Obtain basic skills on Free and Open-Source Software for GIS
CO3	Understand customization techniques in GIS.
CO4	Acquire knowledge on building applications with spatial database.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	✓			✓	✓	
CO2		✓		✓	✓	✓
CO3		✓	✓	✓	✓	✓
CO4		✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Introduction to GIS, Defining GIS and Introduction to Spatial data, thematic characteristics of spatial data, sources of spatial data: census, survey data, air photos, satellite images, field data. Satellite Navigation Systems, Models of the Earth; Geoid and Ellipsoid, Datum and Projections, Spatial and attribute data modelling and Management: Spatial entities - Spatial data structures; Raster and Vector GIS implementation architecture; Desktop GIS, GIS Server, Web GIS applications.

Module 2: Free and Open-Source Software for GIS, Standards and Interoperability, Open Geospatial Consortium Web Map Servers- Web Feature Servers- Metadata standard, XML, Geographic Markup Language.

Module 3: Customization of GIS Overview- the need and benefit of Customization – programming for GIS applications - the enhancement of GIS functionalities through customization – Automation of redundant processes - Data development/update automation – Discuss various case studies that involve customization.

Module 4: Spatial databases, creating a spatially-enabled database, GIS objects, building spatial indexes, spatial queries and spatial functions, Building applications with spatial database, GIS Integration with R and Big Data. Web mapping, Web Mapping Services-Open Layers-Google maps-Yahoo maps and Microsoft map services, Mashups. GeoRSS. Web GIS Implementation: Web Map servers and Data servers

Core Compulsory Readings

1. Heywood.L, Comelius.S and S. Carver ,An Introduction to Geographical Information Systems, Dorling Kinderseley (India) Pvt. Ltd, 2006.

2. Burrough P A 2000 P A McDonnell, Principles of Geographical Information systems, London: Oxford University Press, 2000

Core Suggested Readings

1. Lo.C.P., Yeung. K.W. Albert ,Concepts And Techniques of Geographic Information Systems, Prentice-Hall of India Pvt ltd, New Delhi , 2002.

3. Longley, P.A., Goodchild, M.F., Maguire, D.J. and Rhind, D.W, Geographic Information Systems and Science.Chichester: Wiley. 2nd edition, 2005.

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	**Percentage**
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is a Geographical Information System (GIS) and how is it used in various industries?

2. What are the main components of a GIS, and how do they interact with each other?

3. How can GIS be used to analyze spatial patterns and relationships in data?

4. What are the different methods and techniques used for data collection in GIS?

5. How does GIS contribute to urban planning and land use management?

6. What are the key considerations when creating and managing a spatial database in a GIS?

7. How can GIS be applied in environmental management and conservation efforts?

8. What are the tools and techniques used for geospatial analysis in GIS?

9. How can GIS be used to support decision-making processes in transportation and logistics?

10. What are the current trends and advancements in GIS technology and its applications?

ELECTIVE COURSE – POOL C

MSCSC04DSE25 OPERATION RESEARCH

Credit			Teaching Hours			Assessment Weightage (%)		
L/T	P/I	Total	L/T	P/I	Total	CE	ESE	Total
4	0	4	4	0	4	40	60	100

Lecture/Tutorials, P/I=Practical/Internship, CE =Continuous Evaluation, ESE = End Semester Evaluation

Course Description: The course "Operations Research" offers students an introduction to the field of optimization and decision-making in complex systems. It covers mathematical modelling techniques and optimization methods used to solve real-world problems. Students will learn to formulate problems mathematically, apply linear and nonlinear programming techniques, and use simulation and queuing theory for analysis. The course also explores decision analysis, network optimization, and forecasting methods to aid in making informed decisions. By the end of the course, students will have a solid foundation in operations research and its practical applications in various industries.

Course Objectives:

- Understand the key concepts and techniques of operations research.
- Develop skills to solve complex problems using operations research tools.
- Apply operations research techniques to real-world scenarios for informed decision-making.
- Improve problem-solving and critical thinking abilities through analysis and evaluation of different methods.
- Appreciate the role of operations research in enhancing efficiency and productivity in organizations.

Course Outcomes:

At the end of the Course, the Student will be able to:

SL #	Course Outcomes
CO1	Understand the Linear programming problem solving.
CO2	Solve different LPP and its applications
CO3	Acquire knowledge about Integer Programming and dynamic programming.
CO4	Obtain knowledge about sequencing problem.

Mapping of COs to PSOs

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1		✓	✓		✓	✓
CO2		✓	✓	✓	✓	✓
CO3	✓	✓	✓	✓	✓	✓
CO4	✓	✓	✓	✓	✓	✓

COURSE CONTENTS

Module 1: Linear programming: Formulation, Graphical Solution-2 variables, Development of Simplex Method, Artificial Variable Techniques, Big- M method, Two-Phase method, Reversed Simplex method.

Module 2: Duality in LPP and its formulation, Dual Simplex Method, bounded variable method, Applications of LPP, Transportation problems, Assignment Problem, Traveling Sales persons problem.

Module 3: Integer Programming problem (IPP), Cutting Plane algorithm, Branch and bound method of solving IPP, Dynamic programming problems and its characteristics, Deterministic Dynamic Programming Problem.

Module 4: Sequencing Problem, Processing n jobs through two machines and their mechanics, Processing n jobs through m machines, Processing 2 jobs through m machines, Project scheduling by PERT / CPM, Difference between PERT / CPM, Constructing the network, Critical path analysis, Float of an activity, three time estimated for PERT, project cost by CPM.

Core Compulsory Readings

1. Thaha H.A.- Operation Research, 9THEdn, Pearson

2. Sharm J.K, Mathematical Models in Operation Research, TMGH, 1989.

Core Suggested Readings

1. Trivedi,. Probability, Statistics with Reliability, Queuing and Computer Science Applications, PHI
2. Winston, Operations Research Applications and Algorithms, 4thedn, CENGAGE, 2003Sons, 2007.

TEACHING LEARNING STRATEGIES

- Lecturing, Visualization, Team Learning.

MODE OF TRANSACTION

- Lecture, Seminar, Discussion

ASSESSMENT RUBRICS

Components	Percentage
End Semester Evaluation	**60**
Continuous Evaluation	**40**
Tests	40
Assignment	20
Seminar…..	40

Sample Questions to test Outcomes.

1. What is the purpose of operations research and how does it contribute to decision-making processes?
2. What are the main components of an optimization problem in operations research?
3. How can linear programming be used to solve resource allocation problems?
4. What are the different types of decision analysis techniques used in operations research?
5. How does simulation modeling help in analyzing and predicting outcomes in complex systems?
6. What is the role of sensitivity analysis in operations research and how is it conducted?
7. What are the applications of operations research in supply chain management?
8. How can network analysis techniques be used to optimize transportation and logistics systems?
9. What are the advantages and limitations of using mathematical modelling in operations research?
10. How does operations research contribute to the field of project management and scheduling?

END OF SEMESTER IV