

# LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY

## (An Autonomous Institution)

### DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

### SCHEME OF INSTRUCTION & EXAMINATION [LR-23]

(W. e. f Academic Year 2023-24)

## B.E. I-Semester

S. No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/ D	Contact Hours/Week	Maximum Marks		Duration in Hours	
								CIE	SEE		
<b>MC: Three Week Induction Programme</b>											
<b>Theory Course</b>											
1	U23EN102	MC	Indian Constitution	2	-	-	2	40	60	3	0
2	U23MA101	BSC	Mathematics – I	3	-	-	3	40	60	3	3
3	U23CH101	BSC	Engineering Chemistry	3	-	-	3	40	60	3	3
4	U23EE101	ESC	Basic Electrical Engineering	3	-	-	3	40	60	3	3
5	U23EN101	HSMC	English for Professional Communication	2	-	-	2	40	60	3	2
<b>Practical/ Laboratory Course</b>											
6	U23CH1L1	BSC	Engineering Chemistry Lab	-	-	3	3	25	50	3	1.5
7	U23EE1L1	ESC	Basic Electrical Engineering Lab	-	-	3	3	25	50	3	1.5
8	U23EN1L1	HSMC	Effective Communication and Soft Skills Lab	-	-	3	3	25	50	3	1.5
9	U23ME1L2	ESC	Workshop/Manufacturing Practice Lab	1	-	4	5	50	50	3	3
<b>Total</b>				<b>14</b>	<b>-</b>	<b>13</b>	<b>27</b>	<b>325</b>	<b>500</b>	<b>-</b>	<b>18.5</b>

**L:** Lecture (*Hrs/Wk/Sem*) **T:** Tutorial (*Hrs/Wk/Sem*) **P:** Practical **D:** Drawing (*Hrs/Wk/Sem*)

**CIE:** Continuous Internal Evaluation **SEE:** Semester End Examination

**MC:** Mandatory Course **BS:** Basic Science **ES:** Engineering Science

**HS:** Humanities and Social Sciences **MA:** Mathematics **CH:** Chemistry

**EN:** English

**CE:** Civil Engineering **ME:** Mechanical Engineering. **EE:** Electrical Engineering

**Note:**

1. Each contact hour is a Clock Hour.
2. The duration of the practical class is two hours, however it can be extended wherever necessary, to enable the student to complete the experiment.



**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2024-25)**  
**B.E. III-Semester (Revised)**

S. No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks		Duration in Hours	
								CIE	SEE		
<b>Theory Course</b>											
1	U23MA301	BSC	Mathematics –III ( Probability and Statistics)	3	1	-	4	40	60	3	4
2	U23EC304	ESC	Digital Electronics and Computer Organization	3	-	-	3	40	60	3	3
3	U23EN301	HSMC	English For Technical Communication	2	-	-	2	40	60	3	2
4	U23CS302	PCC	Data Structures	3	-	-	3	40	60	3	3
5	U23CM301	PCC	Python Programming	3	-	-	3	40	60	3	3
<b>Practical/ Laboratory Course</b>											
6	U23EN3L1	HSMC	Soft Skills & Employability Skills Lab	-	-	3	3	25	50	3	1.5
7	U23CM3L1	PCC	Python Programming Lab	0	0	3	3	25	50	3	1.5
8	U23CS3L1	PCC	Data Structures Lab	0	0	3	3	25	50	3	1.5
<b>Bridge Courses *</b>											
9.	U23CS3L2	ESC	C Programming Lab	-	-	2	2	50	-	-	0
10	U23EN3L2	HSMC	Effective Communication Skills Lab	-	-	2	2	50	-	-	0
<b>Total</b>				<b>14</b>	<b>1</b>	<b>9</b> <b>(*13)</b>	<b>23</b> <b>(*27)</b>	<b>275</b> <b>(*375)</b>	<b>450</b>	<b>-</b>	<b>19.5</b>

**\*Bridge Courses Only For Lateral Admitted Students**

**L:** Lecture (Hrs/Wk/Sem) **T:** Tutorial (Hrs/Wk/Sem) **P:** Practical **D:** Drawing (Hrs/Wk/Sem)

**CIE:** Continuous Internal Evaluation **SEE:** Semester End Examination **BSC:** Basic Science Course **ESC:** Engineering Science Course

**PCC:** Program core course **HSMC:** Humanities & Social Science Including Management Course **MA:** Mathematics **EN:** English **CM:**

**CSE-AIML** **PH:** Physics **EC:** Electronics Communication **CD:** CSE-Data Science **AM:** AI&ML

**Note:**

- Each contact hour is a Clock Hour.
- The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2024-25)**

**B.E. IV-Semester (Revised)**

S. No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks		Duration in Hours	
								CIE	SEE		
<b>Theory Course</b>											
1	U23IT402	PCC	Database Management Systems	3	0	0	3	40	60	3	3
2	U23 IT403	PCC	Operating Systems	3	0	0	3	40	60	3	3
3	U23 CS403	PCC	JAVA Programming	3	0	0	3	40	60	3	3
4	U23CM401	PCC	Discrete Mathematics	3	0	0	3	40	60	3	3
5	U23CM402	PCC	Artificial Intelligence	3	1	0	3	40	60	3	4
<b>Practical/ Laboratory Course</b>											
6	U23 IT4L2	PCC	Database Management Systems Lab	0	0	3	3	25	50	3	1.5
7	U23 CS4L1	PCC	JAVA Programming Lab	0	0	3	3	25	50	3	1.5
8	U23CM4L1	PCC	Artificial Intelligence Lab	0	0	3	3	25	50	3	1.5
<b>Total</b>				<b>15</b>	<b>1</b>	<b>9</b>	<b>24</b>	<b>275</b>	<b>450</b>	<b>-</b>	<b>20.5</b>

**L:** Lecture (Hrs/Wk/Sem)    **T:** Tutorial (Hrs/Wk/Sem)    **P:** Practical    **D:** Drawing (Hrs/Wk/Sem)

**CIE:** Continuous Internal Evaluation    **SEE:** Semester End Examination    **BSC:** Basic Science Course    **ESC:** Engineering Science Course    **PCC:** Program core course    **HSMC:** Humanities & Social Science Including Management Course    **MA:** Mathematics    **EN:** English    **CM:** CSE-AIML    **PH:** Physics    **EC:** Electronics Communication    **CD:** CSE-Data Science    **AM:** AI&ML

**Note:**

- Each contact hour is a Clock Hour.
- The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2025-26)**  
**B.E. V-Semester (Tentative)**

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks		Duration In Hours	
								CIE	SEE		
<b>Theory Course</b>											
1	U23MB501	HSMC	Business Economics and Financial Analysis	3	0	0	3	40	60	3	3
2	U23CM501	PCC	Automata Theory, Languages and Computation	3	0	0	3	40	60	3	3
3	U23CS501	PCC	Design & Analysis of Algorithms	3	1	0	3	40	60	3	4
4	U23AM501	PCC	Machine Learning	3	1	0	3	40	60	3	4
5	-	PEC	Professional Elective-I	3	0	0	3	40	60	3	3
<b>Practical/Laboratory Course</b>											
6	U23CS5L1	PCC	Design & Analysis of Algorithms Lab	0	0	3	3	25	50	3	1.5
7	U23CM5L1	PCC	Machine Learning Lab	0	0	3	3	25	50	3	1.5
<b>Internship</b>											
8	U23CM5P1	PROJ	Internship (During Vacation Period After IV Sem)	-	-	2	2	50	-	-	1
<b>Skill Development Course</b>											
9	U23MA5L1	BSC	Aptitude and Reasoning	-	-	2	2	25	50	3	1
<b>Total</b>				<b>15</b>	<b>2</b>	<b>10</b>	<b>25</b>	<b>350</b>	<b>400</b>	<b>-</b>	<b>22</b>

L: Lecture(Hrs/Wk/Sem) T: Tutorial (Hrs/Wk/Sem) P: Practical D: Drawing (Hrs/Wk/Sem)

CIE: Continuous Internal Evaluation SEE: Semester End Examination BSC: Basic Science Course ESC: Engineering Science Course

PCC: Program core course HSMC: Humanities & Social Science Including Management Course MA: Mathematics EN: English CM:

CSE-AIML PH: Physics EC: Electronics Communication CD: CSE-Data Science AM: AI&ML PROJ: Project PEC: Professional Elective Course

**Note:**

- Each contact hour is a Clock Hour.
- The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the

student to complete the experiment.

**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2025-26)**  
**B.E. VI-Semester (Tentative)**

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/ D	Contact Hours/Week	Maximum Marks		Duration in	
								CIE	SEE		
<b>Theory Course</b>											
1	U23AM601	PCC	Robotics Process Automation	2	1	0	3	40	60	3	3
2	U23CM602	PCC	Advanced Machine Learning	3	0	0	3	40	60	3	3
3	U23IT602	PCC	Computer Networks	3	0	0	3	40	60	3	3
4	-	PEC	Professional Elective-II	3	0	0	3	40	60	3	3
5	-	OEC	Open Elective-I	3	-	-	3	40	60	3	3
<b>Practical/Laboratory Course</b>											
6	U23CM6L1	PCC	Advanced Machine Learning Lab	-	-	3	3	25	50	3	1.5
7	U23IT6L1	PCC	Computer Networks Lab	-	-	3	3	25	50	3	1.5
8	U23CS6L2	HSMC	Scripting Languages Lab	-	-	2	2	50	-	-	1
<b>Project</b>											
9	U23CM6P2	PROJ	Mini Project	-	-	6	6	50	50	3	3
<b>Total</b>				<b>14</b>	<b>1</b>	<b>14</b>	<b>29</b>	<b>350</b>	<b>450</b>	<b>24</b>	<b>22</b>

**L:** Lecture (Hrs/Wk/Sem)    **T:** Tutorial (Hrs/Wk/Sem)    **P:** Practical    **D:** Drawing (Hrs/Wk/Sem)

**CIE:** Continuous Internal Evaluation    **SEE:** Semester End Examination    **BSC:** Basic Science Course    **ESC:** Engineering Science Course    **PCC:** Program core course    **HSMC:** Humanities & Social Science Including Management Course    **MA:** Mathematics    **EN:** English    **CM:** CSE-AIML    **PH:** Physics    **EC:** Electronics Communication    **CD:** CSE-Data Science    **AM:** AI&M    **PROJ:** Project    **OEC:** Open Elective Course    **EN:** Engl    **PEC:** Professional Elective Course

**Note:**

- Each contact hour is a Clock Hour.
- The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2026-27)**

**B.E. VII-Semester (Tentative)**

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours / Week	CIE	SEE	Duration in Hours	
<b>Theory Course</b>											
1	U23CM701	PCC	Information Security	3	0	0	3	40	60	3	3
2	U23CM702	PCC	Deep Learning Techniques	3	0	0	3	40	60	3	3
3	-	PEC	Professional Elective – III	3	0	0	3	40	60	3	3
4	-	PEC	Professional Elective – IV	3	0	0	3	40	60	3	3
5	-	OEC	Open Elective – II	3	0	0	3	40	60	3	3
<b>Practical/Laboratory Course</b>											
7	U23CM7L1	PCC	Information Security Lab	-	0	2	2	25	50	3	1
8	U23AM7L1	PCC	Deep Learning Techniques Lab	-	0	3	3	25	50	3	1
8	U23CM7P1	PCC	Major Project Phase – I	-	0	3	3	25	50	3	2
<b>Total</b>				<b>15</b>	<b>0</b>	<b>8</b>	<b>23</b>	<b>275</b>	<b>450</b>	<b>24</b>	<b>19</b>

**L:** Lecture(*Hrs/Wk/Sem*)

**T:** Tutorial(*Hrs/Wk/Sem*)

**P:** Practical

**D:** Drawing(*Hrs/Wk/Sem*)

**CIE:** Continuous Internal Evaluation

**SEE:** Semester End Examination

**CM:** CSE-AIML

**OEC:** Open Elective Course

**PEC:** Professional Elective Course

**PROJ:** Project

**AM:** AI&ML

**Note:**

- Each contact hour is a Clock Hour.
- The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2026-27)**

**B.E. VIII-Semester (Tentative)**

S.No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	CIE	SEE	Duration in Hours	
<b>Theory Course</b>											
1	-	PCC	Professional Elective – V	3	0	0	3	40	60	3	3
2	-	OEC	Open Elective – III	3	0	0	3	40	60	3	3
3	-	OEC	Open Elective – IV	3	0	0	3	40	60	3	3
<b>Practical/Laboratory Course</b>											
8	U23AM8L1	PROJ	Comprehensive Viva	0	0	4	4	100	-	-	2
8	U23CM8P2	PROJ	Major Project Phase – II	0	0	16	16	50	150	-	8
<b>Total</b>				<b>9</b>	<b>0</b>	<b>20</b>	<b>29</b>	<b>270</b>	<b>330</b>	<b>-</b>	<b>19</b>

**L:** Lecture(*Hrs/Wk/Sem*)

**T:** Tutorial(*Hrs/Wk/Sem*)

**P:** Practical

**D:** Drawing(*Hrs/Wk/Sem*)

**CIE:** Continuous Internal Evaluation

**SEE:** Semester End Examination

**CM:** CSE-AIML

**OEC:** Open Elective Course

**PEC:** Professional Elective Course

**PROJ:** Project

**AM:** AI&ML

**Note:**

- Each contact hour is a Clock Hour.
- The duration of the practical class is three hours, however it can be extended wherever necessary, to enable the student to complete the experiment.



**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2023-24)**

**PROFESSIONAL ELECTIVE COURSES (Tentative)**

S. No.	Course Code	Category	Course Title	Scheme of Instructions				Scheme of Examination			CREDITS
				L	T	P/D	Contact Hours/Week	Maximum Marks		Duration in Hours	
								CIE	SEE		
<b>Theory Course</b>											
1	U21CM504	PEC 1	Graph Theory	3	0	0	3	40	60	3	3
	U21CM505		Web and Internet Technologies	3	0	0	3	40	60	3	3
	U21CM506		Foundation Of Data Science	3	0	0	3	40	60	3	3
	U21CM507		Software Engineering	3	0	0	3	40	60	3	3
2	U21CM604	PEC 2	Artificial Neural Networks	3	0	0	3	40	60	3	3
	U21CM605		Mobile Application Development	3	0	0	3	40	60	3	3
	U21CM606		R- For Data Science	3	0	0	3	40	60	3	3
	U21CM607		Compiler Design	3	0	0	3	40	60	3	3
3	U21CM703	PEC 3	Fuzzy Logic	3	0	0	3	40	60	3	3
	U21CM704		Parallel and Distributed Systems	3	0	0	3	40	60	3	3
	U21CM705		Mobile Computing	3	0	0	3	40	60	3	3
	U21CM706		Computer Graphics and 3D Design & Printing	3	0	0	3	40	60	3	3
4	U21CM801	PEC 4	Optimization Techniques	3	0	0	3	40	60	3	3
	U21CM802		Cloud Computing	3	0	0	3	40	60	3	3
	U21CM803		Social Media And Data Analytics	3	0	0	3	40	60	3	3
	U21CM804		Multimedia & Animation	3	0	0	3	40	60	3	3
5	U21CM805	PEC 5	Machine Vision	3	0	0	3	40	60	3	3
	U21CM806		Internet of Things	3	0	0	3	40	60	3	3
	U21CM807		Big Data Analytics	3	0	0	3	40	60	3	3
	U21CM808		Virtual, Augmented and Mixed Reality	3	0	0	3	40	60	3	3

**LORDS INSTITUTE OF ENGINEERING AND TECHNOLOGY**  
**(An Autonomous Institution)**  
**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2023-24)**

**PROFESSIONAL ELECTIVE COURSES (Tentative)**

<b>PE1</b>	<b>PE2</b>	<b>PE3</b>	<b>PE4</b>	<b>PE5</b>
Automata Languages and Computation	Compiler Design	Optimization techniques	Pattern Recognition	Fuzzy Logic
Distributed Databases	Parallel and Distributed Systems	Cloud Computing	Internet of Things	High Performance Computing
Foundation of Data Science	Big Data Analytics	Data Handling and Visualization	Mathematical Modeling and Data Analytics	Data Mining
Software Engineering	Cloud Computing	Information Security	Computer Forensics	Block Chain Technology

**ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING**  
**SCHEME OF INSTRUCTION & EXAMINATION [LR-23]**  
**(W.e.f Academic Year 2021-22)**  
**OPEN ELECTIVE COURSES OFFERED**

S. No.	Course Code	Category	Course Title
1	U23EE508	OEC 1	Non Conventional Energy Systems
	U23EE509		Energy Conservation and Management
	U23CS508		Data Base Management Systems
	U23IT506		Data Structures
	U23ME509		Basics of Mechanical Engineering
	U23ME510		Modern Manufacturing Processes
	U23CE510		Disaster Preparedness and Management
	U23CE511		Civil Engineering Principles and Practices
	U23EC507		Principles of Electronic Communication
	U23EC508		Semi Conductor Devices
	U23MB502		Managerial Communication
	U23MB503		Managerial Science and Theory
	U23SH501		History of Science & Technology
	U23SH502		Economic Policies in India

S. No.	Course Code	Category	Course Title
2	U23EE608	OEC 2	Fundamental of Power Electronics
	U23EE609		Electrical Installation and Safety
	U23CS607		Java Programming
	U23IT606		Operating Systems
	U23ME608		Basics Of 3-D Printing
	U23ME609		Optimization Methods for Engineers
	U23CE607		Construction Materials
	U23CE608		Engineering Geology
	U23EC607		Principles of Data Communication and Network
	U23EC608		Embedded Systems
	U23MB602		Total Quality Management
	U23MB603		Innovation Management
	U23SH601		Indian Music System
	U23SH602		Introduction to Art and Aesthetics

S. No.	Course Code	Category	Course Title
3	U23EE711	OEC 3	Introduction to Electrical Vehicles
	U23EE712		Design estimation and Costing of Electrical Systems
	U23CS711		Data Sciences
	U23IT705		Basics of Artificial Intelligence
	U23ME711		Renewable Energy Resources
	U23ME712		Cooling of Electronic Components
	U23CE711		Environmental Systems
	U23CE712		Urban Transportation System
	U23EC703		IOT and its protocols
	U23EC704		Television and Video Engineering
	U23MB702		Logistics Management
	U23MB703		Management of Start Up's
	U23SH701		Display Devices
	U23SH702		Comparative Study of Literature

S. No.	Course Code	Category	Course Title
4	U23EE804	OEC 4	Smart Building Systems
	U23EE805		Industrial Automation
	U23CS806		Basics of Machine Learning
	U23IT802		Cloud computing
	U23ME806		Hybrid Vehicle Technology
	U23ME807		Power Plant Engineering
	U23CE806		Green Building Technology
	U23CE807		Environmental Impact Assessment
	U23EC805		Fundamentals of Wireless Communication
	U23EC806		Fundamental Digital Design using Verilog HDL
	U23MB802		Entrepreneurship
	U23MB803		E – Marketing
	U23SH801		Corrosion Science and Technology
	U23SH802		Introduction to Philosophical Thoughts

**SEMESTER III**

Course Code	Course Title					Core / Elective	
U23MA301	<b>MATHEMATICS – III (PROBABILITY AND STATISTICS)</b>					Core	
	<b>Branch – CSE-AIML ,IT, AIML , ECE &amp; CSE</b>						
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
--	L	T	D	P			
	3	1	-	-	40	60	4

**Course Objectives**

The objective of the course is to:

1. Introduce the basic concepts of probability and statistics in engineering
2. Provide an overview of concepts of probability and statistics to engineers
3. Provide the knowledge of probability distributions , tests of significance
4. Acquire the concepts of curve fitting, correlation and regression.
5. Familiar with the concept of tests of hypothesis for decision making

**Course Outcomes**

After completing this course, the student will be able to:

1. Determine Probability, Random variables, distributions and its application
2. Apply the knowledge of some standard discrete probability distributions and moments
3. Calculate parameters of standard continuous probability distributions Find the parameters and concepts of correlation, regression and obtain the knowledge of sampling Theory with context to test of hypothesis.
4. Analyze and check the validity of statement using testing of hypothesis for various parameters and goodness of fit.

**Unit-I**

Introduction of Probability, Conditional probability, Baye's Theorem and its applications, Random variables, Types of random variables, Probability mass function and Probability density function, Mathematical expectations.

**Unit-II**

Discrete probability distributions: Binomial and Poisson distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions.

**Unit-III**

Continuous probability distributions, Uniform, Exponential and Normal distributions, Mean, variance and evaluation of statistical parameters for these distributions.

#### **Unit-IV**

Curve fitting by the method of least squares: fitting of straight lines, second degree parabolas and more general curves, Correlation, regression and Rank correlation. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means.

#### **Unit-V**

Small Sample test for single mean, difference of means, test for ratio of variances, Chi-square test for goodness of fit and independence of attributes, Low Rank Matrix, Singular Valued Decomposition (SVD).

#### **Textbooks:**

1. Advanced Engineering Mathematics, R.K.Jain & Iyengar, Narosa Publications.
2. B.S.Grewal, "Higher Engineering Mathematics", Khanna Publishers, 2000.
3. Engineering Mathematics, P.Sivaramakrishna Das & C.Vijaya Kumar, Pearson India Education Services Pvt.Ltd.
4. Engineering Mathematics, SS Sastry, PHI Learning, Private Limited

#### **REFERENCE BOOKS:**

1. Fundamentals of Mathematical Statistics, S.C.Gupta & V.K.Kapoor, S.ChandPub.
2. P.G.Hoel, S.C.Portand C.J.Stone, "Introduction to Probability Theory", Universal Book Stall, 2003.
3. W.Feller, "An Introduction to Probability Theory and its Applications", Vol.1, Wiley, 1968.
4. N.P.Baliand M.Goyal, "A textbook of Engineering Mathematics", Laxmi Publications, 2010.

Course Code	Course Title				Core/Elective		
<b>U23EC304</b>	<b>Digital Electronics &amp; Computer Organization</b>				<b>Core</b>		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	<b>3</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>40</b>	<b>60</b>	<b>3</b>
<b>Course Objectives:</b> The objectives of this course are <ol style="list-style-type: none"> <li>To understand the basic building blocks of digital hardware and various minimization techniques.</li> <li>To analyze and design the Combinational and Sequential circuits.</li> <li>Describe the basic structure and operation of digital computer and understand various memory types</li> </ol>							
<b>Course Outcomes:</b> On Successful completion of this course, student will be able to <ol style="list-style-type: none"> <li>Demonstrate the number system conversions and simplify Boolean functions.</li> <li>Analyze and simplify Boolean expressions using karnaugh-maps, tabulation method and design combinational circuits.</li> <li>Analyze and design various Sequential circuits.</li> <li>To illustrate the operation of digital computer and to understand its organization.</li> <li>Understand the various memory types.</li> </ol>							

**UNIT – I**

**Number Systems:** Number systems, Complements of Numbers, Codes- Weighted and Non-weighted codes and its Properties, Parity check code and Hamming code. Boolean algebra: Basic Theorems and Properties, Switching Functions- Canonical and Standard Form, Algebraic Simplification, Fundamentals of Digital Logic Gates.

**UNIT – II**

**Combination Circuits:** Implementation of logic functions using K-Map Quine- McCluskey Tabular method, Adders: half adder, full adder, Subtractors, Comparators, Multiplexers, Demultiplexers, Encoders and Decoders.

**UNIT-III**

**Sequential circuits:** Basic Architectural Distinctions between Combinational and Sequential circuits, SR Latch, Flip Flops: SR, JK, JK Master Slave, D and T Type Flip Flops. Excitation Table of all Flip Flops, Conversion from one type of Flip-Flop to another.

**UNIT-IV**

**Basic Structure of Computers:** Computer Types, Block diagram of Digital computer, Basic Operational Concepts, Bus Structures, Stored program organization and computer registers, Instruction formats, Input/output Organization: Block diagram of I/O organization, Interrupts, direct memory access. Asynchronous data transfer: strobe control and hand shaking.

**UNIT -V**

**The Memory System:** Basic concepts, Semiconductor RAM memories, Read-Only memories, Speed, Size and Cost, Primary memory ,Auxiliary memory, Associative memory, Cache memories, Virtual Memories, Memory management requirements.

**Text Books:**

1. Morris Mano M. and Michael D. Ciletti, “Digital Design, With an Introduction to Verilog HDL”, Pearson 5 th edition, 2013.
2. 2 RP Jain “ Modern Digital Electronics” , Fourth Edition Mcgraw hill education (India) Pvt Limited, 2003
3. Carl Hamacher, ZvonkoVranesic, SafwatZaky, Computer Organization, 5th Edition, McGraw Hill, 2002.
4. Suggested Reading:
5. Ronald J Tocci, Neal Widmer, Greg Moss, “Digital Systems: Principles and Applications”, Pearson 11th Edition, 2011.
6. Computer Architecture a quantitative approach, Jhon L. Hennessy and David A. Patterson, Fourth Edition Elsevier.
7. Computer Architecture: Fundamentals and principles of Computer Design, Joseph D. Dumas II, BS Publication.



Course code	Course title				Core/Elective		
U23EN301	ENGLISH FOR TECHNICAL COMMUNICATION				--		
Pre-requisites	Contact Hours Per Week				CIE	SEE	Credits
English for Professional Communication	L	T	D	P			
	2	-	-	-	40	60	2
<p><b>Course Objectives:</b> To expose the students to:</p> <ol style="list-style-type: none"> <li>1. Understand the significance of Technical Writing.</li> <li>2. Different types of official correspondence.</li> <li>3. Various styles of technical report writing.</li> <li>4. Designing, creating and developing technical manual.</li> <li>5. Familiarize with the technical features of information transfer.</li> </ol> <p><b>Course Outcomes:</b> On successful completion of the course, the students would be able to:</p> <ol style="list-style-type: none"> <li>1. Apply technical communication skills effectively.</li> <li>2. Adapt different types of official correspondence successfully.</li> <li>3. Construct report writing productively using various techniques.</li> <li>4. Develop the skills of manual writing adequately.</li> <li>5. Interpret the information transfer from verbal to non-verbal data and vice-versa completely</li> </ol>							

### UNIT-I

**Definition and Features of Technical communication:** Definition, Types and Process of Communication, Definition and features of technical communication (precision, relevance, format, style, use of visual aids), Difference between general writing and technical writing, Types of technical communication.

### UNIT-II

**Technical Writing-I (Official correspondence):** Emails, Business letters (all types), Business proposals.

### UNIT-III

**Technical writing-II (Reports):** Definition, Importance, Types of Report - Memo, Letter & Manuscript, Feasibility report, Project report, Progress report, Evaluation report.

### UNIT-IV

**Technical writing- III (Manuals):** Types of manuals, User manual, Product manual, Operations manual

### UNIT-V

**Information Transfer and Presentations:** Non-verbal (bar diagram, flow chart, pie chart, tree diagram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oral and visual presentations.

### Reference Books:

1. Kumar, Kulbhushan, (2019), *Effective Communication Skills*, Khanna Publishing House.

2. Raman, Meenakshi & Sharma, Sangeeta. (2017). *Technical Communication: Principles and Practice*, OUP (3<sup>rd</sup> Ed.), New Delhi.
3. Rizvi, Ashraf, M. (2018). *Effective Technical Communication* (2nd Ed.). Tata McGraw Hill Education. New Delhi.
4. Sharma, R. C., & Mohan, Krishna. (2017). *Business Correspondence and Report Writing: A Practical Approach to Business & Technical Communication* (4<sup>th</sup> Ed.). Tata McGraw Hill Education. New Delhi.
5. Tyagi, Kavita & Misra, Padma, (2011), *Advanced Technical Communication*. New Delhi, PHI Learning.

Course Code	Course Title				Core/Elective		
<b>U23CS302</b>	<b>DATA STRUCTURES</b>				<b>Core</b>		
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS	3	-	-	-	40	60	3

**Course Objectives:**

Develop ability to

1. Develop skills to design and analyze simple linear and nonlinear data structures, such as stacks, queues and lists and their applications.
2. Gain programming skills to implement sorting and searching algorithms
3. Strengthen the ability to identify and apply the suitable data structures for the given real world problem.
4. Gain knowledge in practical applications of data structures
5. Understand essential for future programming and software engineering courses.

**Course Outcomes:**

At the end of the course, student would be able to

1. Implement various data structures using arrays, linked lists
2. Develop ADT necessary for solving problems based on Stacks and Queues
3. Implement binary trees, general tree structures, advanced search trees, heaps, graphs.
4. Implement hash functions and handle collisions.
5. Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem

**UNIT – I****Introduction Data Structures and Algorithms:**

Introduction to data structures, classification of data structures, operations on data structures; Algorithm Specification, Recursive algorithms, Data Abstraction, Performance analysis- Time Complexity and Space Complexity, Asymptotic Notation-Big O, Omega, and Theta notations.

**UNIT – II****Stacks and Queues:**

Stacks: Stack ADT, definition and operations, Implementations of stacks using array, applications of stacks, Arithmetic expression conversion and evaluation.

Queues: Queue ADT, definition and operations, Implementation of queues using Arrays, applications of linear queue, circular queue and double ended queue (deque).

**UNIT – III**

**Linked Lists:** Introduction, Singly linked list, representation of a linked list in memory, operations on a Singly Linked List, Implementation of Singly Linked List.

Doubly Linked Lists; Operations on Doubly Linked List, Implementation of Doubly Linked List, Circular linked list, Implementation of Stack and Queue using linked list.

**UNIT – IV**

**Searching Techniques:** Linear search and Binary Search algorithms.

**Sorting Techniques:** Bubble Sort, Insertion sort, Selection Sort, Merge Sort, and Quick Sort. Comparison among sorting techniques.

**Hashing:** Static Hashing, Hash Tables, Hash Functions, Overflow Handling, Theoretical Evaluation of Overflow Techniques

**UNIT – V**

**Trees:** Introduction, Binary Trees, Tree Traversals, Threaded Binary Trees, Binary Search Tree, Heap Tree, AVL Tree.

**Graphs:** Graph Abstract Data Type, Representation of Graph, Graph Traversals -DFS and BFS, Spanning Tree, Prim's and Kruskal's Algorithms.

**Suggested Readings:****Text Books:**

1. "Fundamentals of Data Structures in C", Ellis Horowitz, Sartaj Sahani, Susan Anderson Freed, Computer Science Press, 2004
2. S. Lipschutz, "Data Structures", Tata McGraw Hill Education, 1st Edition, 2008.

**Reference Books:**

3. D. Samanta, "Classic Data Structures", PHI Learning, 2nd Edition, 2004.
4. Mark A Weiss, "Data Structures and Algorithm Analysis In C", Second Edition (2002), Pearson
5. "Data Structures and Algorithms in C++", second Edition by Michael T. Goodrich and [Roberto Tamassia](#).

Course Code	Course Title				Core /Elective		
U23CM301	PYTHON PROGRAMMING				Core		
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS	3	-	-	-	40	60	3

**Course Objectives:**

Develop ability to

1. Learn about Python programming language syntax, semantics, and the runtime environment
2. Familiarized with universal computer programming concepts like data types, containers
3. Acquire general computer programming concepts like conditional execution, loops & functions
4. Grasp the general coding techniques and object-oriented programming
5. Analyze about basic library modules.

**Course Outcomes:**

At the end of the course, student would be able to

1. Develop essential programming skills in computer programming concepts like data types, containers.
2. Apply the basics of programming in the Python language.
3. Solve coding tasks related conditional execution, loops.
4. Acquire coding tasks related to the fundamental notions and techniques used in object oriented programming
5. Write basic programs related to basic library modules.

**UNIT- I**

**Introduction to Python:** Data Types: Declaring and using Numeric data types: int, float, bool, complex, string data type and string operations, standard input and output functions, type conversion

**Python Program Flow Control:** Conditional blocks using if, else and elif, Control blocks using for loop, while loop. Loop manipulation using pass, continue, break and else.

**UNIT- II**

**Functions:** Creating, parameters and return values, Using Keyword Arguments and Default Parameter Values, Using Global Variables and Constants, recursive functions, lambda functions.

**Python Data Structures:** Lists- basic list operators, replacing, inserting, removing an element; searching and sorting lists; tuples.

**UNIT-III**

**Dictionaries-** literals, adding and removing keys, accessing and replacing values; traversing dictionaries. Sets and Strings operations.

**Introduction to Object Oriented Programming:** Class, object, attributes and methods; defining classes, inheritance, polymorphism, abstract classes, Exception handling.

**UNIT-IV**

**Using Databases in Python:**

Python MySQL Database Access, Create Database Connection, CREATE, INSERT, READ, UPDATE and DELETE Operation, DML and DDL Operation with Databases, Performing Transactions Handling, Database Errors.

#### **UNIT- V**

##### **Python for Data Analysis:**

**Numpy:** Introduction to numpy, Creating arrays, Using arrays and Scalars, Indexing Arrays, Array Transposition, Universal Array Function, Array Processing, Array Input and Output.

**Pandas:** What is pandas? Where it is used? Series in pandas, Index objects, Reindex , Drop Entry, Selecting Entries, Data Alignment, Rank and Sort Summary, Statics Missing Data, Index Hierarchy.

##### **Text Books:**

1. Reema Thareja, “Python Programming using Problem Solving Approach”, ISBN-13:978-0-19-948017-3, Oxford University Press, 2017
2. Vamsi kurama, “Python Programming: A modern approach”, ISBN-978-93-325-8752- 6, Pearson,2018.

##### **Reference Books:**

3. Mark Lutz , “Learning python”, ISBN: 1-56592-464-9, Orielly, 4th edition, 1999 .
4. W.Chun, “Core python programming”, ISBN-13: 978-0132269933, Pearson, 2nd edition, 2016.
5. Kenneth Lambert, ” Fundamentals of Python: First Programs”, ISBN-13: 978-1337560092, Cengage Learning publishers, First Edition,2012.
6. Allen B. Downey, “Think Python: How To Think Like A Computer Scientist”, ISBN-13: 978-1491939369. O’Reilly, 2<sup>nd</sup> Edition, 2016.
7. R Nageswara Rao, “Core Python Programming”, Dreamtech press, 2017 Edition.
8. Mike Mc Grath "Python in easy steps: Makes Programming Fun", Kindle Edition, 2017.

Course code	Course title					Core/Elective	
U23EN3L1	Soft Skills & Employability Skills Lab [Common to all branches]					Core	
Pre-requisites	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	3	25	50	1.5
<p><b>Course Objectives:</b> To expose the students to:</p> <ol style="list-style-type: none"> <li>1. Apply soft skills at professional level.</li> <li>2. Foster leadership skill with a mature outlook for effective functioning at work front.</li> <li>3. Develop confidence through interpersonal skills.</li> <li>4. Exhibit their ability and skills to write Resume/CV and cover letter</li> <li>5. Boost skills of group discussion and interview.</li> </ol> <p><b>Course Outcomes:</b> On successful completion of the course the students would be able to:</p> <ol style="list-style-type: none"> <li>1. Utilise soft skills at professional level effectively.</li> <li>2. Function efficiently in multidisciplinary settings by using leadership skills.</li> <li>3. Build confidence through interpersonal skills utterly.</li> <li>4. Write Resume/CV and cover letter comprehensively.</li> <li>5. Enhance the skills of group discussion and interview perfectly</li> </ol>							

### List of Activities

#### 1. Soft Skills

Introduction to Soft Skills and Types; Time Management, Team work

#### 2. Leadership Skill

Decision Making, Critical Thinking, Conflict Resolution, Adaptability Skills

#### 3. Interpersonal Skills

Stress Management, Emotional Intelligence, Motivation, Presentation Skills

#### 4. Job Skills

Resume/CV writing, Cover letter writing

#### 5. Interview Skills

Dynamics of Group Discussion, Types; Interview, Types, Interview Etiquettes, Mock Interviews,

#### **Suggested Readings:**

1. Bhardwaj, Kumkum, (2019), *Fundamentals of Business Communication*, Wiley, India
2. Kapoor Shikha, (2020), *Personality development and Soft Skills-Preparing for Tomorrow*, Wiley India
3. Koneru, Arun, (2017), *Professional Communication*, Tata McGraw-Hill Publishing Company. Ltd, New Delhi
4. Mitra K. Barun. (2016). *Personality Development and Soft Skills*. Oxford University Press.
5. Raman Meenakshi & Sharma Sangeeta, (2017), *Technical Communication: Principles and Practice*, OUP (3rd Ed.). New Delhi.
6. Sharma, Prashant, (2019). *Soft Skills-Personality development for Life Success*, BPB Publications
7. Tyagi, Kavita & Misra, Padma. (2011). *Advanced Technical Communication*, PHI Learning, New Delhi.

Course Code	Course Title				Core		
<b>U23CS3L1</b>	<b>DATA STRUCTURES LAB</b>				<b>Core</b>		
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
<b>PPS Lab</b>	-	-	-	<b>3</b>	<b>25</b>	<b>50</b>	<b>1.5</b>
<p><b>Course Objectives:</b>            Develop ability to</p> <ol style="list-style-type: none"> <li>1. Understand essential concepts of simple linear and nonlinear data structures.</li> <li>2. Analyze and implement programming skills to implement sorting and searching algorithms</li> <li>3. Apply the suitable data structures for the given real world problems.</li> <li>4. Acquire knowledge in practical applications of data structures.</li> <li>5. Provide solutions for various graphical concepts.</li> </ol> <p><b>Course Outcomes:</b>            At the end of the course, student would be able to</p> <ol style="list-style-type: none"> <li>1. Write programs in various data structures using arrays and linked lists.</li> <li>2. Develop ADT necessary for solving problems based on Stacks and Queues</li> <li>3. Evaluate binary trees, general tree structures, advanced search trees, heaps, graphs.</li> <li>4. Apply hash functions and handle collisions</li> <li>5. Implement various kinds of sorting techniques and apply appropriate techniques for solving a given problem</li> </ol>							

### List of Experiments:

1. Implementation of Stacks and Queues using Arrays.
2. Solving Towers of Hanoi problem
3. Implementation of Circular Queue.
4. Solving tic-tac -toe problem
5. Implementation of Infix to Postfix Conversion, Postfix Expression Evaluation.
6. Implementation of Singly Linked List
7. Implementation of Doubly Linked List.
8. Implementation of Circular Linked List.
9. Implementation of Stacks, Queues using Linked Lists.
10. Implementation of Binary Search and Hashing
11. Implementation of Operations on Binary Tree (Delete Entire Tree, Copy Entire Tree, Mirror Image, Level Order, Search for a Node etc.)
12. Implementation of Tree Traversals on Binary Trees.
13. Implementation of Binary Search Tree. (Insertion, Deletion and Search operations)
14. Implementation of operations on AVL Trees.



15. Implementation of Traversal on Graphs.
16. Implementation of Selection, Merge, Quick and InsertionSort.
17. Implementation of Prims and Kruskals Algorithm.

**Suggested Readings:**

- 1.S. Lipschutz, “Data Structures”, Tata McGraw Hill Education, 1st Edition, 2008.
- 2.D. Samanta, “Classic Data Structures”, PHI Learning, 2nd Edition, 2004.
3. Mark A Weiss, Data Structures and Algorithm Analysis In C, SecondEdition (2002), Pearson

Course Code	Course Title				Core/Elective		
U23CM3L1	PYTHON PROGRAMMING LAB				Core		
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
PPS	-	-	-	3	25	50	1.5
<p><b>Course Objectives:</b></p> <p>Develop ability to</p> <ol style="list-style-type: none"> <li>1. Elucidate problem solving through python programming.</li> <li>2. Learn the use of functions in python programming.</li> <li>3. Analyze solutions using Object Oriented concepts.</li> <li>4. Design programs using File Handling Functions.</li> <li>5. Create Mini-projects using various libraries.</li> </ol> <p><b>Course Outcomes:</b></p> <p>At the end of the course, student would be able to</p> <ol style="list-style-type: none"> <li>1. Summarize the fundamental concepts of python programming.</li> <li>2. Outline the control statements and functions by writing python program.</li> <li>3. Demonstrate file handling operations and packages.</li> <li>4. Interpret object oriented programming in python.</li> <li>5. Apply the suitable libraries to solve simple problems.</li> </ol>							

### List of Programming Exercises:

- 1 a) Write a python program for Python Variables, Executing Python from the Command Line, Editing Python Files, Python Reserved Words.
  - b) Write a python program to add two numbers.
  - c) Write a program to demonstrate different number data types in python.
  - d) Write a program to perform different arithmetic operations on numbers in python.
- 2) a) Write a python program to print a number is positive/negative using if-else.
  - b. Write a python program to find largest number among three numbers.
  - c. Write a Python program to swap two variables
  - d) Python Program to print all Prime Numbers in an Interval
- 3) a) Write a python program to check whether the given string is palindrome or not.
  - b. Write a program to create, concatenate and print a string and accessing substring from a given string.
  - c) Functions: Passing parameters to a Function, Variable Number of Arguments, Scope, and Passing Functions to a Function.
- 4) a) Create a list and perform the following methods
  - 1) insert() 2) remove() 3) append() 4) len() 5) pop() 6)clear()

- b) Create a dictionary and apply the following methods
  - 1) Print the dictionary items 2) access items 3) useget () 4) change values 5) use len()
- c) Create a tuple and perform the following methods
  - 1) Add items 2) len() 3) check for item in tuple 4)Access items
  
- 5) a). OOP concepts: Classes, File Organization, Special Methods, Inheritance, Polymorphism, Special Characters, Character Classes, Quantifiers, Dot Character, Greedy Matches, Matching at Beginning or End, Match Objects, Compiling Regular Expressions.
- b) Write a python Program to call data member and function using classes and objects
  
- 6) a). Write a program to double a given number and add two numbers using lambda ()
- b) Write a program for filter () to filter only even numbers from a given list.
- c) Write a Python Program to Make a Simple Calculator.
  
- 7) a). Demonstrate a python code to print try, except and finally block statements
- b) Write a python program to open and write “hello world” into a file and check the access permissions to that file?
- c) Python program to sort the elements of an array in ascending order and Descending order
  
- 8) a) Write a python program to open a file and check what are the access permissions acquired by that file using os module.
- b) Write a program to perform basic operations on random module.
  
- 9) Write a python program to practice some basic library modules
  - a) Numpy
  - b) SciPy
- 10) Introduction to basic concept of GUI Programming and Develop desktop based application with python basic Tkinter() Module?
- 11) Write a python program to create a package (college), subpackage (alldept), modules(it,cse) and create admin function to module?
- 12). Write a python program to create a package (Engg), sub package(years), modules (sem) and create staff and student function to module?

**Suggested Readings:**

1. Mrak Summerfield, “Programming in Python: A complete introduction to the Python language” Addison-Wesley professional:2009
2. Martin C. Brown, “PYTHON: the complete reference” McGraw-Hill,2001
3. W.J. Palm III, introduction to MATLAB 7 for engineers, McGraw-Hill international Edition 2005.
4. Wesley J Chun. “Core Python Applications Programming” Prentice Hall. 2012
5. Allen B Downey, “Think Python”, O’ Reilly. 2012

**SEMESTER IV - SYLLABUS**

Course Code	Course Title					Core/Elective	
<b>U23IT402</b>	<b>Database Management Systems</b>					<b>Core</b>	
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	40	60	3

**Course Objectives:**

The objectives of this course is to impart knowledge

1. Understand the role of database management system in an organization and learn the database concepts.
2. Design databases using data modeling and Logical database design techniques.
3. Construct database queries using relational algebra and calculus and SQL.
4. Understand the concept of a database transaction and related concurrent, recovery facilities.
5. Understand the concepts of Triggers and Stored Procedures.

**Course Outcomes**

At the end of the Course, Student would be:

1. Design ER-models to represent simple database application scenarios and Construct database queries using SQL.
2. Construct database queries using relational algebra and calculus.
3. Recognize and identify the use of normalization and functional dependency in database design.
4. Apply the concept of a database transaction and related concurrent, recovery facilities
5. Apply and relate how to evaluate a set of queries in query processing.

**UNIT - I**

**CONCEPTUAL MODELING INTRODUCTION:** Introduction to Data bases: Purpose of Database systems, view of data, data models, Database languages, Database users, various components of overall DBS architecture, various concepts of ER model, basics of Relational Model.

**SQL QUERY – BASICS:**

SQL – Data Definition commands, Queries with various options, Data manipulation commands, Views, Joins, views, integrity and security.

**UNIT – II****RELATIONAL APPROACH**

Relational algebra and calculus: Relational algebra, selection and projection, set operations, renaming, joins, division, examples of algebra queries, relational calculus: Tuple relational calculus, Domain relational calculus, expressive power of algebra and calculus.

**UNIT - III**

**INTRODUCTION TO NoSQL:** Introduction, Overview and History of NoSQL Databases – The Definition of the Four Types of NoSQL Databases, differences between SQL and NoSQL .

**NORMALIZATION:** Pitfalls of RDBD, Lossless join decomposition, functional dependencies, Armstrong axioms, normalization for relational databases 1st, 2nd and 3rd normal forms, BCNF, Basic definitions of MVDs and JDs, 4th and 5th normal forms.

#### **UNIT - IV**

**TRANSACTION MANAGEMENT:** Transaction processing: Transaction concept, transaction State, implementation of atomicity and durability, concurrent executions, serializability, recoverability. Concurrency Control: Lock-based protocols, timestamp-based protocols, validation-based protocols, multiple granularities, multi-version schemes, deadlock handling.

#### **UNIT – V**

**DATA STORAGE:** Overview of physical storage media, magnetic disks, storage access, file organization, organization of records in files.

Indexing and Hashing: Basic concepts, types of indexing, difference between B and B+ Indexing, static hashing, Dynamic Hashing.

#### **Suggested Readings:**

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", McGraw-Hill, 6th Edition, 2017.
2. Ramez Elmasri, Shamkant B. Navathe, "Fundamental Database Systems", Pearson Education, 6th Edition, 2014.
3. Raghu Ramakrishnan, "Database Management System", Tata McGraw-Hill Publishing Company, 3rd Edition, 2007.
4. Hector Garcia Molina, Jeffrey D. Ullman, Jennifer Widom, "Database System Implementation", Pearson Education, United States, 1st Edition, 2000.
5. Peter Rob, Corlos Coronel, "Database System, Design, Implementation and Management", Thompson Learning Course Technology, 5th Edition, 2003.

Course Code	Course Title				Course Category		
U23IT403	Operating Systems				PCC		
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
DLD	L	T	D	P			
	3	-	-	-	40	60	3

**Course Objectives:**

1. To understand the services provided by and the design of an operating system.
2. To understand the structure and organization of the file system.
3. To understand what a process is and how processes are synchronized and scheduled.
4. To understand different approaches to memory management.
5. To Understand Virtual machine concepts, calls for managing processes, memory and the file system

**Course Outcomes:**

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

1. Understand the fundamental concepts and Functions of operating system.
2. Analyze various scheduling algorithms.
3. Understand deadlock, prevention and avoidance algorithms.
4. Compare and contrast various memory management schemes.
5. Understand the functionality of file systems and perform administrative tasks on Linux Servers

**UNIT-I**

**Introduction:** Computer System organization & Architecture, Operating System Structure & Operations, Process, Memory and Storage Managements, Protection and Security, Distributed and Special-Purpose Systems, Computing Environments.

**System Structures:** System calls, Types of System Calls, System Programs.

**Process Concept:** Overview, Process Scheduling, Operations on Processes, Inter process communication, Communication in Client/Server Systems, Operating System Examples.

**UNIT-II**

**Process Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, Priority, Round Robin)

**Synchronization:** Background, The Critical-Section Problem, Peterson's Solution, Synchronization, Dining Philosophers problem Monitors.

**Deadlocks:** System Model, Deadlock characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

**UNIT-III****Memory-Management Strategies:**

Background, Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.

**Virtual Memory Management:** Background, Demand paging, Copy-on-write, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory.

**Storage Management:** File System, File Concept, Access Methods, Directory Structure, File-System Mounting, File sharing, Protection.

#### UNIT-IV

**Implementing File Systems:** File System-Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery, Log-Structured File Systems, NFS.

**Secondary Storage Structure:** Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, RAID Structure, Stable-Storage Implementation, Tertiary-Storage Structure.

#### UNIT-V

**Protection and Security:** Goals of Protection, Principles of Protection, Domain of protection, Access Matrix, Implementation of Access Matrix, Access control, Revocation of access rights,

**System Security:** The security problem, program Threats, System and System Network Threats, Cryptography as a Security tool, User Authentication, Implementing Security Defences, firewalling to protect Systems and Networks, Case Studies- Linux System.

#### Text Book

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Principles, ninth Edition, John Wiley & Sons Publication,2012
2. A.Tanenbaum-Modern Operation Systems. Third edition, Pearson Education, 2008.

#### Reference Books

1. William Stallings - Operating Systems, Fifth Edition, Pearson Education, 2005.
2. Ida M.Flynn, Understanding Operating Systems, Sixth Edition, Cengage, 2011
3. Operating Systems: Principles and Practice, Thomas Anderson and Michael Dahlin, Recursive Books, 2014.
4. The Design of Unix Operating System, Maurice Bach, Prentice Hall, 1988.

Course Code	Course Title				Core/Elective		
<b>U23CS403</b>	<b>JAVA Programming</b>				<b>Core</b>		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
<b>PPS</b>	<b>3</b>	<b>-</b>	<b>-</b>	<b>-</b>	<b>40</b>	<b>60</b>	<b>3</b>

**Course Objectives:**

Develop ability to

1. Understand fundamentals of object-oriented programming in Java which includes defining classes, invoking methods, difference between applet and application programs, using class libraries
2. Create Java application programs using sound OOP practices such as interfaces, exception handling, multi-threading.
3. Understand fundamentals of object-oriented programming in Java.
4. Define classes, invoking methods, difference between applet and application programs, using class libraries
5. Use Collection framework, AWT and event handling to solve real world problems.

**Course Outcomes**

At the end of the Course, Student would be:

1. Achieve proficiency in object-oriented concepts and also learns to incorporate the same into the Java programming language.
2. Create Java application programs using sound OOP practices e.g. Inheritance, interfaces and proper program structuring by using packages, access control specifiers.
3. Understand and Implement the concepts of Exception Handling in JAVA.
4. Develop the ability to solve real-world problems through software development in high-level programming language using Large APIs of Java as well as the Java standard class library.
5. Understand File, Streams, Input and Output Handling in java.

**UNIT – I**

**Object Oriented Programming:** Benefits of Object Oriented Programming.

**Introduction to Java:** Java buzzwords, bytecode. Java Programming Fundamentals, data types, variables, arrays, operators, expressions, control statements, concepts of classes, objects, constructors, methods, access control, overloading methods and constructors, introducing access control, static, final, exploring string class.

**Principles of OOPS:** Data Abstraction, Data Encapsulation, Polymorphism, and Inheritance.

**UNIT – II**

**Interfaces:** Defining an interface, implementing interfaces, extending interface. **Packages:** Defining, Creating and Accessing a Package, importing packages

**Exception handling:** Benefits of exception handling, classification, checked exceptions and unchecked



exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, built in exceptions, creating own exception subclasses

### UNIT – III

**Multithreading:** Java Thread Model, The Main Thread, creating a Thread, creating multiple threads, using is Alive() and join(), thread priorities, synchronization, inter thread communication, deadlock

**Collections:** Overview of Java Collection frame work, commonly used Collection classes.

**Other Utility classes:** String Tokenizes, Scanner Java Input/output:exploring java.io, Java I/O classes and interfaces, File, Stream classes, byte stream, character stream, serialization.

### UNIT – IV

**GUI Programming with java:** The AWT class hierarchy, MVC architecture. Applet Revisited: Basics, architecture and skeleton, simple applet program.

**Event Handling:** Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces. Handling mouse and keyboard events, Adapter classes.

**Database Programming using JDBC:** Introduction to JDBC, JDBC Drivers & Architecture, CRUD operation Using JDBC.

### UNIT – V

**Exploring Swing:** JLabel, ImageIcon, JTextField, the Swing buttons, JTabbedPane, JScrollPane, JList, JComboBox.

**Servlet:** Life cycle, using tomcat, simple servlet, servlet API, javax. servlet package, reading servlet parameters, javax. servlet. http package, handling HTTP requests and responses

#### Text Books:

1. Herbert Scheldt, “The Complete Reference Java, 7th Edition, Tata McGraw Hill,2006.
2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning,2002.

#### Reference Books:

3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 5th Edition, McGraw Hill Publishing,2010.
4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education /PHI.

Course Code	Course Title				Core/Elective		
<b>U23CM401</b>	<b>DISCRETE MATHEMATICS</b>				<b>Core</b>		
Prerequisite	Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	<b>3</b>	<b>0</b>	<b>-</b>	<b>-</b>	<b>40</b>	<b>60</b>	<b>3</b>

**Course Objectives:**

Develop ability to

1. Understand concepts of Mathematical Logic, mechanisms of inference rules for propositional and predicate logic and their applications
2. Understand the concepts of Sets, Relations, Functions and their applications.
3. Learn the concepts of Algebraic Structures, basics of counting, Principles of inclusion/exclusion and the pigeonhole methodology.
4. Understand Generating Functions, Recurrence Relations and various ways of solving them.
5. Understand basic definitions and properties of graphs and their applications in computer science and engineering.

**Course Outcomes:**

At the end of the course, student would be able to

1. Distinguish between Propositional Logic and Predicate Logic, deriving valid proofs of inference and checking the validity of inferences.
2. Illustrate by examples the basic terminology of sets, relations, functions and algebraic structures along with their associated operations.
3. Demonstrate basics of counting, principles of permutations, combinations, applying inclusion/exclusion principle and the pigeonhole methodology in solving counting problems.
4. Demonstrate Generating functions, write recurrence relations and apply various techniques solving recurrence relations.
5. Transform a problem in computer science and engineering as a graph to solve it efficiently using Concepts of graph theory.

**UNIT-I**

**Mathematical Logic:** Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence implication, Normal forms, Quantifiers, universal quantifiers. Predicates: Predicative logic, Free & Bound variables, Rules of inference, Consistency, proof of contradiction, Automatic Theorem Proving.

**UNIT-II**

**Relations:** Properties of Binary Relations, equivalence, transitive closure, compatibility and partial ordering

relations, Lattices, Hasse diagram. Functions: Inverse Function Composition of functions, recursive Functions, Lattice and its Properties.

### **UNIT-III**

**Elementary Combinatorics:** Basis of counting, Combinations & Permutations, with repetitions, Constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, the principles of Inclusion – Exclusion. Pigeon hole principles and its application.

### **UNIT-IV**

**Recurrence Relation:** Generating Functions, Function of Sequences Calculating Coefficient of generating function, Recurrence relations, Solving recurrence relation by substitution and Generating funds. Characteristics solution of inhomogeneous Recurrence Relation.

### **UNIT-V**

**Graph Theory:** Basic Concepts, Representation of Graph, Isomorphism, Sub graphs, Spanning Trees, Planar Graphs, Multi graphs, Euler circuits, Hamiltonian graphs, Chromatic Numbers. Graph Theory and Applications.

### **Text Books::**

1. Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay, R. Manohar, McGraw Hill education (India) Private Limited. (UNITS - I , II )
2. Discrete Mathematics for Computer Scientists & Mathematicians, Joe L. Mott, Abraham Kandel, Theodore P. Baker, Pearson , 2nd ed. (Units - III, IV, V )

### **Reference Books**

3. Elements of Discrete Mathematics- A Computer Oriented Approach- C L Liu, D P Mohapatra. Third Edition, Tata McGrawHill.
4. Discrete Mathematics for Computer Scientists & Mathematicians, J.L. Mott, A.Kandel, T.P. Baker, PHI.
5. Discrete Mathematics and its Applications, Kenneth H. Rosen, Fifth Edition. TMH.
6. Discrete Mathematical Structures Theory and Application-Malik & Sen, Cengage.

Course Code	Course Title				Core/Elective		
<b>U23CM402</b>	<b>ARTIFICIAL INTELLIGENCE</b>				<b>Core</b>		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Data structures (graphs), Basics of probability	<b>3</b>	<b>1</b>	-	-	<b>40</b>	<b>60</b>	<b>04</b>

**Course Objectives**

The course will introduce the students to

- To understand the various characteristics of Intelligent agents
- To learn the different search strategies in AI
- To learn to represent knowledge in solving AI problems
- To understand the different ways of designing software agents
- To know about the various applications of AI.

**Course Outcomes**

After successful completion of the course the students will be able to

- Use appropriate search algorithms for any AI problem
  - Represent a problem using first order and predicate logic
  - Provide the apt agent strategy to solve a given problem
  - Design software agents to solve a problem
  - Design applications for NLP that use Artificial Intelligence.

**UNIT I INTRODUCTION**

Introduction–Definition - Future of Artificial Intelligence – Characteristics of Intelligent Agents– Typical Intelligent Agents – Problem Solving Approach to Typical AI problems.

**UNIT II PROBLEM SOLVING METHODS**

Problem solving Methods - Search Strategies- Uninformed - Informed - Heuristics - Local Search Algorithms and Optimization Problems - Searching with Partial Observations - Constraint Satisfaction Problems – Constraint Propagation - Backtracking Search - Game Playing - Optimal Decisions in Games – Alpha - Beta Pruning - Stochastic Games

**UNIT III KNOWLEDGE REPRESENTATION**

First Order Predicate Logic – Unification – Forward Chaining-Backward Chaining – Resolution – Knowledge Representation - Ontological Engineering-Categories and Objects – Events - Mental Events and Mental Objects - Reasoning Systems for Categories - Reasoning with Default Information

**UNIT IV SOFTWARE AGENTS**

Architecture for Intelligent Agents – Agent communication – Negotiation and Bargaining – Argumentation among Agents – Trust and Reputation in Multi-agent systems.

**UNIT V APPLICATIONS**

AI applications – Language Models – Information Retrieval- Information Extraction – Natural Language Processing - Machine Translation – Speech Recognition – Robot – Hardware – Perception – Planning – Moving

**TEXT BOOKS:**

1. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, Third Edition, 2009.
2. I. Bratko, —Prolog Programming for Artificial Intelligence, Fourth edition, Addison-Wesley Educational Publishers Inc., 2011.

**REFERENCES:**

1. M. Tim Jones, —Artificial Intelligence: A Systems Approach(Computer Science), Jones and Bartlett Publishers, Inc.; First Edition, 2008
2. Nils J. Nilsson, —The Quest for Artificial Intelligence, Cambridge University Press, 2009.
3. William F. Clocksin and Christopher S. Mellish, Programming in Prolog: Using the ISO Standard, Fifth Edition, Springer, 2003.
4. Gerhard Weiss, —Multi Agent Systems, Second Edition, MIT Press, 2013.
5. David L. Poole and Alan K. Mackworth, —Artificial Intelligence: Foundations of Computational Agents, Cambridge University Press, 2010.

Course Code	Course Title				Core/Elective		
<b>U23IT4L2</b>	<b>Database Management Systems Lab</b>				<b>Core</b>		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	3	25	50	1.5

**Course Objectives:**

1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation
3. To understand the basic concepts and the applications of database systems.
4. Be acquainted with the basics of transaction processing and concurrency control.
5. Learn the concepts of Views, Stored Procedure and Triggers.

**Course Outcomes**

After completing this course, the student will be able to:

1. Design database schema for a given application and apply normalization
2. Gather skills in using SQL commands for data definition and data manipulation.
3. Demonstrate creation and usage of Views and Stored Procedures using SQL.
4. Develop solutions for database applications using procedures, cursors and triggers
5. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

**LIST OF EXPERIMENTS**

**Scenario:** Product-Sales database: South wind database is a sample database used by Organization. The database contains the sales data for South wind Traders; it is foods export-import Company. Using this schema to demonstrate the how customer can choose and order products, how orders are placed and how those products get delivered to the customer. Products: This Entity will have all the products details where suppliers will supply products based on customers demand. Supplies: This Entity will supply the products demanded by the customers. Shippers: This Entity will take the orders from suppliers and deliver to customers. Employees: Employees will monitor the orders placed by customers. Invoices: This Entity will take care of billing process based on customer order. Etc...Identify some more entities and find out relationship between them. Product-sales the above process involves many steps like

1. Analyzing the problem and identifying the Entities and Relationships,
2. E-R Model
3. Relational Model
4. Normalization
5. Creating the database
6. Querying.

**Experiment 1: E-R Model**

Analyze and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc. Identify the primary keys for all the entities. Identify the other keys like Foreign Key and constraints like NULL, NOT NULL, CHECK etc. Example to create for products, customers, suppliers, orders, , employees, order details, categories, among others. Students should submit E-R diagrams using the above tables.

**Experiment 2: DDL**

How to create tables, altering the database or tables, dropping tables if not required. You will also try truncate, rename commands etc. Data Definition Language (DDL) : create , alter, drop.

**Experiment 3: DML**

Data Manipulation Language Commands (DML) commands are used to for managing data within schema objects. Exercising the commands using DML: insert, delete, update on the following tables : products, customers, suppliers, orders, , employees, order details, categories.

- INSERT – insert data into a table.
- UPDATE – updates existing

**Experiment 4: Querying**

Data within a table.

- DELETE – deletes single or all records from a table.

Data Query Language – Select Populate all the tables designed in experiment: 2 with appropriate data.

Practice queries on Aggregate functions like count, max , min ,avg ,sum Practice queries like nested queries/co-related queries using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, groupby ,having etc.

Joins: Join , Left Outer Join, Right Outer Join, Self Join

**Experiment 5 : Querying(continued...)**

Some example to practice the queries:

1. Display all the order details of given a customer.
2. Display all the products.
3. Get the highest sold product from given supplier ID
4. List all products grouped by category
5. List the products, whose products unit price is greater then all the products of average.
6. List Details of order and customer of each order
7. List the products which were sold in year 1997
8. Display the total amount for each order
9. Display Order Details for given an order ID
10. Order Details: product name and unit price for given order ID Exercising Simple to complex
11. Queries using joins, nested and co-related queries.

**Experiment 6 : Programs on pl/sql**

1. Write a PL/SQL program to swap two numbers.
2. Write a PL/SQL program to find the largest of three numbers
3. Write a PL/SQL program to find the total and average of 6 subjects and display the grade.
4. Write a PL/SQL program to find the sum of digits in a given number.
5. Write a PL/SQL program to display the number in reverse order.
6. Write a PL/ SQL program to check whether the given number is prime or not.
7. Write a PL/SQL program to find the factorial of a given number.

**Experiment 7 : Stored Procedures :**

1. Create a stored procedure, Alter and Drop a procedure, IN, OUT, IN & OUT parameters
2. Create a Procedure to display order details of given customer ID like ordered, order Date , Required Date, Shipped Date
3. Create a procedure to accept a customer ID and display the customer order history (product name and how much quantity ordered for that particular product)  
Ex: product name, Total quantity he/she ordered.
4. Create a procedure to display Ten Most Expensive Products Columns should be displayed  
Product name & Unit price

**Experiment 8: Views**

1. Create a view to display the current product list which is available (not discontinued)
2. Create a view to display the products by category
3. Display product name, quantity Per Unit, units In Stock, Discontinued
4. Create a view as —Invoicesl to display all the information from order, customer, and shipper for each Order Details

**Experiment 9: Triggers**

Demonstrate Create Trigger, Alter Trigger, Drop Trigger, Row Level, Table Level triggers, Before Insert, After Insert, Before Update, After Update, Before Delete, After Delete

**Experiment 10: Case study: Book Publishing Company**

A publishing company produces scientific books on various subjects. The books are written by authors who specialize in one particular subject. The company employs editors who, not necessarily being specialists in a particular area, each take sole responsibility for editing one or more publications.

A publication covers essentially one of the specialist subjects and is normally written by a single author. When writing a particular book, each author works with on editor, but may submit another work for publication to be supervised by other editors. To improve their competitiveness, the company tries to employ a variety of authors, more than one author being a specialist in a particular subject for the above case study, do the following:

- a. Analyze the data required.
- b. Normalize the attributes.

Create the logical data model using E-R diagrams.

**Experiment 11: Case Study: General Hospital**

A General Hospital consists of a number of specialized wards (such as Maternity, Pediatric, Oncology, etc). Each ward hosts a number of patients, who were admitted on the recommendation of their own GP and confirmed by a consultant employed by the Hospital. On admission, the personal details of every patient are recorded. A separate register is to be held to store the information of the tests undertaken and the results of a prescribed treatment. A number of tests may be conducted for each patient. Each patient is assigned to one leading consultant but may be examined by another doctor, if required. Doctors are specialists in some branch of medicine and may be leading consultants for a number of patients, not necessarily from the same ward. For the above case study

For the above case study, do the following.

- a. Analyze the data required.
- b. Normalize the attributes.

Create the logical data model using E-R diagrams.

**Suggested Readings**

1. Raghurama Krishnan, Johannes Gehrke, —Database Management Systemsl, Tata McGraw Hill, 3rd Edition, 2008.
2. Silberschatz, Korth, —Database System Conceptsl, McGraw Hill, V edition, 2005.
3. Rick F. Vander Lans, —Introduction to SQLl, Pearson education, 2007.
4. B. Rosenzweig and E. Silvestrova, —Oracle PL/SQLl, Pearson education, 2004.
5. Dr. P. S. Deshpande, —SQL & PL/SQL for Oracle 10gl, Black Book, Dream Tech, 2006.
6. M. Mc Laughlin, —Oracle Database 11g PL/SQL Programmingl, TMH, 2017



Course Code	Course Title					Core/Elective	
<b>U23CS4L1</b>	<b>JAVA Programming LAB</b>					<b>Core</b>	
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	3	25	50	1.5

**Course Objectives:**

Develop ability to

1. Build software development skills using java programming for real world applications.
2. Implement frontend and backend of an application
3. Create Java application programs using sound OOP practices such as interfaces,exception handling multi-threading.
4. Understand fundamentals of object-oriented programming in Java.
5. Implement classical problems using java programming.

**Course Outcomes**

After completing this course, the student will be able to:

1. Develop Java applications using the concepts of Inheritance, interfaces, packages, access control specifiers.
2. Implement the concepts of Exception Handling in java Applications.
3. Read and write data using different Java I/O streams.
4. Create graphical user interfaces and Applets by applying the knowledge of EventHandlering.
5. Create robust applications using Java standard class libraries and retrieve data from adatabase with JDBC.

**List of Experiments:**

1. (a) Write a Java program to illustrate the concept of class with method overloading  
(b) Write a Java program to illustrate the concept of class with method overriding
2. (a) Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java.util)  
  
(b) Write a Java program to illustrate the concept of Single level and Multi levelInheritance.
3. (a) write a Java program to demonstrate polymorphism  
(b) Write a Java program to demonstrate the Interfaces & Abstract Classes.
4. (a) Write a Java program to implement the concept of exception handling.  
(b) Write a Java program to illustrate the concept of threading using Thread Classand runnable Interface.
5. (a) Write a Java program to illustrate the concept of Thread synchronization.
6. (a) Write a Java program that correctly implements producer consumer problemusing the concept

- of inter
- thread communication.
- (b) Write a Java program that reads a file name from the user, and then displays inform action about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the
- length of the file in bytes.
7. (a) Write a Java program to illustrate the concept of I/O Streams  
(b) Write a Java program to implement serialization concept
8. (a) Write a Java applet program to implement Colour and Graphics class  
(b) Write a Java applet program for handling mouse & key events
9. (a) Write a Java applet program to implement Adapter classes  
(b) Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits
- and for the +, -, \*, % operations. Add a text field to display the result.
10. (a) Write an example for JDBC prepared statement with ResultSet  
(b) Write a Java Program to get primary key value (auto-generated keys) from inserted queries using JDBC
11. (a) Write a Java Program to create a simple JList  
(b) Write a Java Program to create a simple checkbox using JCheckBox
12. (a) Write a Java Program to create a checkbox and Item Listener to it.  
(b) Write Servlet application for following
- i. Html & Servlet Communication
  - ii. Auto refresh a page
  - iii. Demonstrate session tracking
  - iv. Select record from database
  - v. Application for login page
  - vi. Insert record into database
  - vii. Count the visits on webpage
  - viii. Insert teacher record in Database

**Suggested Readings:**

1. Herbert Scheldt, "The Complete Reference Java, 7th Edition, Tata McGraw Hill, 2006.
2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning, 2002.
3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 5th Edition, McGraw Hill Publishing, 2010.
4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education.

Course Code	Course Title				Core/Elective		
<b>U23CM4L1</b>	<b>Artificial Intelligence Lab</b>				<b>Core</b>		
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	3	25	50	1.5

**Course Objectives:**

The objective of this lab is to get an overview of the various Artificial Intelligence techniques and can able to demonstrate those using python.

- To introduce students to the basic concepts of AI Searching techniques
- To develop skills for solving practical problems.
- To gain experience neural networks.

**Course Outcomes:**

After the completion of the course the student can able to:

- After learning the AI concepts the student must be able to design and implement AI solutions searching techniques using AI.
- Able to know about facts of querying.
- Be capable of confidently applying tree mechanism using AI with Neural Network
- Be capable of performing experiments in Machine Learning using real-world data.
- Be capable to implement classifiers and Regression algorithm

**List of Experiments:**

1. Write a Program to Implement Breadth First Search
2. Write a Program to Implement Depth First Search
3. Write a Program to Implement Tic-Tac-Toe game
4. Write a Program to Implement 8-Puzzle problem
5. Write a Program to Implement Water-Jug problem
6. Write a Program to Implement Travelling Salesman Problem
7. Write a Program to Implement Tower of Hanoi
8. Write a Program to Implement Monkey Banana Problem
9. Write a Program to Implement Missionaries-Cannibals Problems
10. Write a Program to Implement N-Queens Problem using Python.
11. Write a program to train and validate the following classifiers for given data (scikit-learn):
  - i. Decision Tree
  - ii. Multi-layer Feed Forward neural network
- b. Implementation of Gaussian Naive Bayes classifier using scikit-learn ( Any two classifiers).
12. Write a program to Implementation of Linear Regression using python( Any Two Algorithm).

**Suggested Readings:**

1. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, Third Edition, 2009.

2. M. Tim Jones, —Artificial Intelligence: A Systems Approach(Computer Science), Jones and Bartlett Publishers, Inc.; First Edition, 2008