ST. PHILOMENA'S COLLEGE (AUTONOMOUS)

Affiliated to University of Mysore Accredited by
NAAC with 'B^{++}' Grade
Bannimantap, Mysore, Karnataka, India-570015

DEPARTMENT OF Master of Computer Application

The Board of Studies in MCA which met on 07/11/2023 has

Approved the syllabus and pattern of examination for I and II semesters for the Academic Year 2023-25

BOS COMMITTEE MEMBERS

Sl.No	Name	Designation &Address	Remarks
1	Dr. Suresha	Professor , Department of Studies in Computer Science, University of Mysore, Manasagangotri , Mysore.	University Nominee By VC Email: sureshasuvi@gmail.com Phone No.9449810894
2	Dr. D. S Guru	Professor , Department of Studies in Computer Science, University of Mysore, Manasagangotri ,Mysore.	External Member Email: dsguruji@yahoo.com dsg@compsci.uni-mysore.ac.in Phone No.9620228005
3	Dr. R. Kumar	Head, Dept. of Computer Science (PG), Kristu Jayanti College (Autonomous), Bengaluru.	External Member Email:rkumar@kristujayanthi.com Phone No.9902933747
4	Dr. Anil Kumar K.M	Professor, Dept. of Computer Science &Engineering, Sri Jayachamarajendra College of Engineering (SJCE), JSS Science and Technology University, Mysore.	External Member Email: anilkm@jssstuniv.in anilkm@sjce.ac.in Phone No.9945633380
5	Mrs. Reena Mol V.U	Assistant Professor, Department of MCA, St. Philomena's College(Autonomous), Mysore.	Internal Member & Chairperson Email: reenamolvu@stphilos.ac.in Phone No.8660083765
6	Mr.Vimal Chandran	Technical Lead, Wells Fargo, 129, Marathahalli - Sarjapur Outer Ring Rd, Devarabisanahalli, Bellandur, Bengaluru, Karnataka 560103	Industry Expert Email:vimalchandran.cj@gmail.com Phone No.8553279616
7	Mr. Noel Nitin Dsouza	Senior Project Engineer Wipro Limited, 146/147, Metagalli Industrial Area, Metagalli,Mysuru.	Alumni Email: noel.dsouza@wipro.com Phone No. 8722328244

Master of Computer Application

2 Years/ 4 Semesters

COURSE CURRICULUM REGULATIONS 2023

AS PER UGC GUIDELINES ON ADOPTION OF CHOICE BASED CREDIT SYSTEM

1. Course Title

The course shall be called **'Master of Computer Applications Post-graduate Degree Course'.**

Duration : 2 YEARS

Number of Semesters : 4

A candidate joining the course shall pursue prescribed courses of studies.

2. Definitions

Course

Every course offered will have three components associated with the teaching-learning process namely

(i) Lecture – L (ii) Tutorial- T (iii) Practical - P, where

L stands Lecture session.

T stands Tutorial session consisting participatory discussion / Self-study / desk work/ brief seminar presentations by students and such other novel methods that make a student to absorb and assimilate more effectively the contents delivered in the Lecture classes.

P stands Practical session and it consists of Hands on experience / Laboratory Experiments / Field Studies / Case studies that equip students to acquire the much required skill component.

In terms of credits, every one hour session of L amounts to 1 credit per semester and a minimum of two hour session of T or P amounts to 1 credit per semester, over a period of one semester of 16 weeks for teaching-learning process. The total duration of a semester is 20 weeks inclusive of semester-end examination.

A course shall have either or all the three components. That means a course may have only lecture component, or only practical component or combination of any two or all the three components.

The total credits earned by a student at the end of the semester upon successfully completing the course are L + T + P. The credit pattern of the course is indicated as L: T: P.

If a course is of 4 credits then the different credit distribution patterns in L: T: P format could be

4 : 0 : 0,	1 : 2 : 1,	1 : 1 : 2,	1 : 0 : 3,	1 : 3 : 0,
2 : 1 : 1,	2 : 2 : 0,	2 : 0 : 2,	3 : 1 : 0,	3 : 0 : 1,
0 : 2 : 2,	0 : 4 : 0,	0 : 0 : 4,	0 : 1 : 3,	0 : 3 : 1

The concerned BoS will choose the convenient credit pattern for every course based on the requirement. However, generally, a course shall be of 3 or 4 credits.

Different courses of study are labeled and defined as follows:

Core Course

A course which should compulsorily be studied by a candidate as a core requirement is termed as a Core course.

A Core course may be a **Soft Core** if there is a choice or an option for the candidate to choose a course from a pool of courses from the main discipline / subject of study or from a sister / related discipline / subject which supports the main discipline / subject. In contrast to the phrase Soft Core, a compulsory core course is called a **Hard Core** Course.

Elective Course

Generally a course which can be chosen from a pool of courses and which may be very specific or specialized or advanced or supportive to the discipline / subject of study or which provides an extended scope or which enables an exposure to some other discipline / subject/domain or nurtures the candidate's proficiency/ skill is called an Elective Course. Elective courses may be offered by the main discipline/ subject of study or by sister / related discipline / subject of study. A Soft Core course may also be considered as an elective.

An elective course chosen generally from an unrelated discipline / subject, with an intention to seek exposure is called an **open elective**.

An elective course designed to acquire a special / advanced knowledge, such as supplement study / support study to a project work, and a candidate studies such a course on his own with an advisory support by a teacher is called a **Self -Study.**

A core course offered in a discipline / subject may be treated as an elective by other discipline / subject and vice versa.

Project work/Dissertation work is a special course involving application of knowledge in solving / analyzing /exploring a real life situation / difficult problem. A project work up to 4 credits is called Minor Project work. A project work of 6 to 8 credits is called Major Project Work. Dissertation work can be of 10-12 credits. A Project work/Dissertation work may be a hard core or a soft core as decided by the BoS concerned.

3. Eligibility for Admission

Candidates possessing a degree of University of Mysore, or of any other University, equivalent there to and complying with the eligibility criteria:

Passed BCA/ Bachelor Degree in Computer Science Engineering or equivalent Degree.

OR

Passed B.Sc./ B.Com./ B.A. with Mathematics at 10+2 Level or at Graduation Level (with additional bridge Courses as per the norms of the concerned University).

Admission to MCA course shall be open for candidates who have passed the Bachelor degree

examinations with not less than 50% of the marks in the aggregate of all the years of the Degree examinations. However, in the case of candidates from Karnataka belonging to SC/ST and Category-I, the aggregate percentage of marks in the qualifying examinations shall not be less than 45%. Provided that for admission to MCA, the candidate shall have passed Bachelor Degree with not less than 50% of marks with Mathematics / Statistics / Computer Science / Computer Programming / Computer Application / Business Mathematics / Business Statistics as one of the optional or electives at degree level. Provided further that in respect of candidates who have studied and passed one of the subjects specified in the first proviso in the Pre- university course with 50% of marks in that subject shall also be considered for admission.

However, in the case of candidates belonging to SC/ST and Category-I, 45% of marks in that subject shall also be considered for admission.

4. Scheme of Instructions

4.1 A Master's Degree program is of 4 semesters - two years duration. A candidate can avail double the duration (in one stretch) to complete Master's Degree (including blank semesters, if any). Whenever a candidate opts for blank semester(s)/DROP in a course or in courses or is compelled to DROP a course or courses as per the provision of the regulation, he/she has to study the prevailing courses offered by the department as per the prevailing scheme, when he/she continues his/her study.

4.2 A candidate has to earn a minimum of 76 credits, for successful completion of a Master's degree with a distribution of credits for different courses as given in the following table.

Course Type	Credits
Hard Core	40
Soft Core	A minimum of 32 but not exceeding 52
Open Elective	A minimum of 04

Every course including project work/Dissertation work, practical work, field work, seminar, self-study elective should be entitled as hard core or soft core or open elective by the BoS concerned.

4.3 A candidate can enroll for a maximum of 24 credits per semester with the approval of the concerned department.

4.4 Only such candidates who register for a minimum of 20 credits per semester in the first two semesters and complete successfully 76 credits in 4 successive semesters shall be considered for declaration of ranks, medals and are eligible to apply for student fellowship, Scholarship , free ships and hostel facilities.

4.5 In excess to the minimum of 76 credits for master's degree in the concerned discipline / subject of study, a candidate can opt to complete a minimum of 20 extra credits to acquire **add on proficiency diploma** in that particular discipline / subject along with the masters' degree. In such of the cases where in, a candidate opts to earn at least 4 extra credits in different discipline / subjects in addition to a minimum of 76 credits at masters level, then an **add on proficiency certification** will be issued to the candidate by listing the courses studied and grades earned.

4.6 A candidate admitted to Master's program can exercise an option to exit with Bachelor Honors degree / PG diploma after earning 40 credits successfully.

5. Continuous Assessment, Earning of Credits and Award of Grades

The evaluation of the candidate shall be based on continuous assessment. The structure for evaluation is as follows:

5.2.1 The first component (C_1), of assessment is for 15 marks. This will be based on test, assignment and seminar. During the first half of the semester, the first 50% of the syllabus will be completed. This shall be consolidated during the 8^{th} week of the semester. Beyond 8^{th} week, making changes in C_1 is not permitted

5.2.2 The second component (C_2), of assessment is for 15 marks. This will be based on test, assignment, and seminar. The continuous assessment and scores of second half of the semester will be consolidated during the 16^{th} week of the semester. During the second half of the semester the remaining units in the course will be completed.

5.2.3 The outline for continuous assessment activities for Component-I (C_1) and Component-II (C_2) will be proposed by the teacher(s) concerned before the commencement of the semester and will be discussed and decided in the respective Departmental Council. The students should be informed about the modalities well in advance. The evaluated courses/assignments during component I (C_1) and component II (C_2) of assessment are immediately returned to the candidates after obtaining acknowledgement in the register maintained by the concerned teacher for this purpose.

5.2.4 During the $18^{th} – 20^{th}$ week of the semester, a semester-end examination of 2 hours duration shall be conducted for each course. This forms the third/final component of assessment (C_3) and the maximum marks for the final component will be 70.
A candidate will be assessed on the basis of:

a) Knowledge of relevant processes
b) Skills and operations involved
c) Results / products including calculation and reporting.

Setting questions papers and evaluation of answer scripts.

I Questions papers in three sets shall be set by the internal examiner for a course. Whenever there are no sufficient internal examiners, the chairmanof BoE shall get the questions papers set by external examiners.

II The Board of Examiners shall scrutinize and approve the questionpapers and scheme of valuation.

III (i) There shall be single valuation for all theory papers by internal examiners. In case, the number of internal examiners falls short, external examiners may be invited.

(ii) The examination for Practical work/ Field work/Project work will be conducted jointly by two internal examiners. However the BoE on its discretion can also invite external examiners if required.

(iii) If a course is fully of (L=0):T (P=0) type, then the examination for C_3 Component will be as decided by the BOS concerned.

IV Challenge valuation

A student who desires to apply for challenge valuation shall obtain a Xerox copy of the answer script by paying the prescribed fee within 10 days after the announcement of the results. He / She can challenge the grade awarded to him/her by surrendering the grade card and by submitting an application along with the prescribed fee to the Registrar (Evaluation) within 15 days after the announcement of the results. This challenge valuation is only for C_3 component.

The answer scripts for which challenge valuation is sought for shall be sent to another external examiner. The marks awarded will be the higher of the marks obtained in the challenge valuation and in maiden valuation.

5.2.5 In case of a course with only practical component a practical examination will be conducted with two examiners (ref: 6.2.3 III (ii)). A candidate will be assessed on the basis of a) knowledge of relevant processes b) Skills and operations involved c) Results / products including calculation and reporting. If external examiner does not turn up then both the examiners will be internal examiners.

The duration for semester-end practical examination shall be decided by the departmental

council.

5.2.6 If **X** is the marks scored by the candidate out of 70 in C_3 in theory examination, if **Y** is the marks scored by the candidate out of 70 in C_3 in Practical examination, and if **Z** is the marks scored by the candidate out of 70 in C_3 for a course of (L=0):T:(P=0)type that is entirely tutorial based course, then the final marks M in C_3 is decided as per the following table.

L.T.P distribution	Find mark M in C_3
L:T:P	$\frac{[(L+T)*X]+[(T+P)*Y]}{L+2T+P}$
L:(T=0):P	$\frac{(L*X)+(P*Y)}{L+P}$
L:T:(P=0)	X
L:(T=0):(P=0)	X
(L=0):T :P	Y
(L=0): (T=0):P	Y
(L=0): T :(P=0)	Z

5.2.7 The details of continuous assessment are summarized in the following Table.

Component	Syllabus in a course	Weightage	Period of Continuous assessment
C_1	First 50% (2 units of total units)	15%	First half of the semester. To be consolidated by 8th week
C_2	Remaining 50% (Remaining units of the course)	15%	Second half of the semester. To be consolidated by 16th week
C_3	Semester-end examination (All units of the course)	70%	To be completed during 18th-20th Week.
Final grades to be announced latest by 24th week			

5.2.8 A candidate's performance from all 3 components will be in terms of scores, and the sum of all three scores will be for a maximum of 100 marks (15 + 15 + 70).

5.2.9 Finally awarding the grades should be completed latest by 24th week of the semester

5.3 Minor/ Major Project Evaluation

Right from the initial stage of defining the problem, the candidate has to submit the progress reports periodically and also present his/her progress in the form of seminars in addition to the regular discussion with the guide. Components of evaluation are as follows.

Component – I(C_1): Periodic Progress and Progress Reports (15%) Component – II (C_2): Results of Work and Draft Report (15%)

Component– III (C_3): Final Viva-voce and evaluation (70%). The report evaluation is for 40% and the Viva-voce examination is for 30%

5.4 In case a candidate secures less than 30% in C_1 and C_2 put together in a course, the candidate is said to have DROPPED that course, and such a candidate is not allowed to appear for C_3 in that course. In case a candidate's class attendance in a course is less than 75% or as stipulated by the University, the candidate is said to have DROPPED that course, and such a candidate is not allowed to appear for C_3 in that course.
Teachers offering the courses will place the above details in the Department Council meeting during the last week of the semester, before the commencement of C_3, and subsequently a notification pertaining to the above will be brought out by the Chairman of the Department before the commencement of C_3 examination. A copy of this notification shall also be sent to the office of the Registrar & Registrar (Evaluation).

5.5 In case a candidate secures less than 30% in C_3, he/she may choose DROP/MAKE-UP option.
In case a candidate secures more than or equal to 30% in C_3, but his/her grade (G) = 4, as per section 6.9 below, then he/she may be declared to have been conditionally successful in this course, provided that such a benefit of conditional clearance based on G=4 shall not be availed for more than 8 credits for the entire program of Master's Degree of two years.
In case a candidate secures less than 30% in C_3, he/she may choose DROP/MAKE-UP option. The candidate has to exercise his/her option to DROP immediately within 10 days from the date of notification of results.
A MAKE UP examination for odd semester courses will be conducted along with next regular odd semester examinations and for even semester courses along with a next regular even semester examinations. If a candidate is still unsuccessful, he/she may opt for DROP or again take up MAKE UP examination; however, not exceeding double the duration norm in one stretch from the date of joining the course.

5.6 A candidate has to re-register for the DROPPED course when the course is offered again by the department if it is a hard core course. The candidate may choose the same or an alternate core/elective in case the dropped course is soft core / elective course. A candidate who is said to have DROPPED project work has to re-register for the same subsequently with in the stipulated period. **The details of any dropped course will not appear in the grade card.**

5.7 The tentative / provisional grade card will be issued by the Registrar (Evaluation) at the end of every semester indicating the courses completed successfully. This statement will not contain the list of DROPPED courses.

5.8 Upon successful completion of Bachelors Honors / Master's degree a final grade card consisting of grades of all courses successfully completed by the candidate will be issued by the Registrar (Evaluation).

5.9 The grade and the grade point earned by the candidate in the subject will be as given below.

Marks	Grade	Grade Point (GP = V x G)
30-39	4	V*4
40-49	5	V*5
50-59	6	V*6
60-64	6.5	V*6.5
65-69	7	V*7
70-74	7.5	V*7.5
75-79	8	V*8
80-84	8.5	V*8.5
85-89	9	V*9
90-94	9.5	V*9.5
95-100	10	V*10

Here, P is the percentage of marks (P= $[(C_1+C_2)+M]$ secured by a candidate in a course which is rounded to nearest integer. V is the credit value of course. G is thegrade and GP is the grade point.

5.10 A candidate can withdraw any course within in ten days from the date of notification of final results. Whenever a candidate withdraws a paper, he/she has to register for the same course in case it is hard core course, the same course or an alternate course if it is soft core/open elective.

5.11 A DROPPED course is automatically considered as a course withdrawn.
Overall cumulative grade point average (CGPA) of a candidate after successful completion the required number of credits (76) is given by

$$\mathbf{C}GPA = \Sigma \mathbf{GP} / \mathbf{Total\ number\ of\ credits}$$

6. Classification of results

The final grade point (FGP) to be awarded to the student is based on CGPAsecured by the candidate and is given as follows.

CGPA	FGP	
	Numerical Index	Qualitative Index
4 <= CGPA < 5	5	SECOND CLASS
5 <= CGPA < 6	6	
6 <= CGPA < 7	7	FIRST CLASS
7 <= CGPA < 8	8	
8 <= CGPA < 9	9	DISTINCTION
9 <= CGPA < =10	10	

Overall percentage=10*CGPA or is said to be 50% in case CGPA<5

7. Attendance and Conduct

The Course is a full-time course and students **SHALL NOT** take up any employment/course, part time or full time during their study. Students found violating this rule shall be removed from the course. Minimum attendance of 75% of actual working hours is required in each paper. A student who does not satisfy the requirements of attendance and conduct shall not be permitted to appear for the examination in the concerned subject.

8. Medium of Instruction

The medium of instruction shall be English. However, a candidate will be permitted to write the examinations either in English or in Kannada. This rule is not applicable to languages.

9. Provision for appeal

If a candidate, is not satisfied with the evaluation of C1 and C2 components, he / she can approach the grievance cell with the written submission together with all facts, the assignments, test papers etc, which were evaluated. He/she can do so before the commencement of semester-end examination. The grievance cell is empowered to revise the marks if the case is genuine and is also empowered to levy penalty as prescribed by the university on the candidate if his/her submission is found to be baseless and unduly motivated. This cell may recommend taking disciplinary/corrective action on an evaluator if he/she is found guilty. The decision taken by the grievance cell is final.

For every program there will be one grievance cell. The composition of the grievance cell is as follows.

1.The Registrar (Evaluation) ex-officio Chairman / Convener
2. One senior faculty member (other than those concerned with the evaluation of the course concerned) drawn from the department/discipline and/or from the sister departments/sister disciplines.
3. One senior faculty members / subject experts drawn from outside the University department.

MCA Program Outcome:

After successful completion of MCA degree, the graduates will be able to:

- Apply the knowledge of Computer Science, Mathematics, Statistics and computing fundamentals to design and develop applications to provide creative solutions to various real life applications.
- Integrate and apply efficiently the contemporary IT tools and design applications with appropriate considerations for any specific need on societal and environmental aspects.
- Involve in perennial learning for a continued career development and progress as a computer professional upholding the ethics, social, cultural and cyber regulations.
- Function effectively both as a team leader and team member on multi-disciplinary projects to demonstrate computing and management skills and also to effectively present technical information in verbal and written reports.
- Apply the inherent skills with absolute focus to function as a successful entrepreneur.

Program Specific Outcome (PSO)

- Understand the concepts and applications in the field of Computing Sciences like Web designing and development, Mobile application development, Network and communication technologies.
- Apply the learning from the courses and develop applications for real world problems.
- Understand the technological developments in the usage of modern design and development tools to analyze and design for a variety of applications.
- Communicate in both oral and written forms, demonstrating the practice of professional ethics and the concerns for social welfare.

Please Note : Subjected to change pending Approval.

St. PHILOMENAS COLLEGE (AUTONOMOUS)

MASTER OF COMPUTER APPLICATION (MCA)

CHOICE BASED CREDIT SYSTEM 2023-24

MCA Course Structure and Syllabi

Minimum Credits Required for MCA Degree

I to IV Semesters	Hard Core Course (HC)		Soft core Course (SC)		Open Elective Course(OE)		Total	
	Numbers	Credits	Numbers	Credits	Number	Credits	Numbers	Credits
	10	40	8	32	1	4	19	76

Credit allotment Semester wise

Semester	Minimum Credit	Maximum Credit
I	20	24
II	24	24
III	20	24
IV	12	24
TOTAL	**76**	**96**

Scheme and Syllabus

SEMESTER I

LIST OF HARD CORE SUBJECTS

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Data Structure and Problem Solving using C	3:0:0	3	3
2	Operating Systems and Linux	3:1:0	4	5
3	Computer Networks	3:0:0	3	3
4	Computer Organization and Architecture	3:1:0	4	5
5	Data Structure and Problem Solving using C Lab	0:0:1	1	2
6	Computer Networks Lab	0:0:1	1	2

LIST OF SOFT CORE SUBJECTS (At most Two)

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Mathematical foundations for Computer Science	3:1:0	4	5
2	Business Systems	3:1:0	4	5
3	Data Mining and Data Warehousing	3:0:1	4	5
4	Probability and Statistics	3:1:0	4	5

SEMESTER II
LIST OF HARD CORE SUBJECTS

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Java Programming	2:1:0	3	4
2	Advanced Database Management Systems	3:0:0	3	3
3	Java Programming Lab	0:0:1	1	2
4	Advanced Database Management Systems Lab	0:0:1	1	2

LIST OF SOFT CORE SUBJECTS (At most Three)

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Analysis & Design of Algorithms	2:1:1	4	6
2	Big Data Analytics	3:1:0	4	5
3	Foundation of Cyber Security	3:1:0	4	5
4	Introduction to Cloud Computing	3:1:0	4	5
5	Artificial Intelligence and Robotics	3:1:0	4	5
6	Web Technologies	3:1:0	4	5

OPEN ELECTIVE (Any One)

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
Offered by Non-Computer Science Department		4-0-0	4	4

SEMESTER III

LIST OF HARD CORE SUBJECTS

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Software Engineering	3:1:0	4	5
2	Python Programming	3:0:0	3	3
3	Python Programming Lab	0:0:1	1	2

LIST OF SOFT CORE SUBJECTS (At most Four)

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Machine Learning	2:1:1	4	6
2	Cloud Computing Architecture	3:1:0	4	5
3	Machine Learning in Cyber Security	3:1:0	4	5
4	Data Science	3:1:0	4	5
5	Digital Image Processing	3:0:1	4	5
6	Internet of Things	3:1:0	4	5
7	Mobile Application Development	2:1:1	4	6
8	Ethical Hacking	1:1:2	4	7

SEMESTER IV
LIST OF HARD CORE SUBJECTS

Sl. No.	Course Title	L-T-P	Credits	Credits
1	Major Project	0:2:6	8	14 hrs /week

LIST OF SOFT CORE SUBJECTS (At most Four)

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Research Methodology	3:1:0	4	5
2	Block Chain Technology	3:1:0	4	5
3	Theory of Languages and Automata	3:1:0	4	5
4	Cloud Security	3:1:0	4	5
5	Software Testing	3:0:1	4	5
6	E-commerce and E-governance	3:1:0	4	5
7	Communication Skills and Professional Management	3:1:0	4	5
8	Soft Computing	3:1:0	4	5

LIST OF OPEN ELECTIVE SUBJECTS OFFERED FROM THE DEPARTMENT

Sl. No.	Course Title	L-T-P	Credits	Contact Hours
1	Artificial Intelligence	3:1:0	4	5
2	Python Programming	3:1:0	4	5
3	Web Technologies	3:1:0	4	5

Semester I
HARD CORE SUBJECTS

Course Code:

Course Name: Data Structure and Problem Solving Using C

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
3 Hrs./Week	-	3	30	-	-	70	100

Course Description:

An overview of data structure concepts, arrays, stack, queues, trees, and graphs. Discussion of various implementations of these data objects, programming styles, and run-time representations. Course also examines algorithms for sorting, searching and some graph algorithms.

Course Outcome:

Upon successful completion of this course, the student will be able to:

COURSE OUTCOMES	
CO1	Design and analyze programming problem statements.
CO2	Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
CO3	Apply mathematical abstraction to solve problems.
CO4	Demonstrate various methods of organizing large amounts of data..
CO5	Analyze algorithms and to determine algorithm correctness and time efficiency class
CO6	Demonstrate linear data structures linked list, stack and queue (apply)
CO7	Implement tree, graph, hash table and heap data structures (apply)
CO8	Apply brute force and backtracking techniques (apply)
CO9	Demonstrate greedy and divide-conquer approaches (apply)
CO10	Implement dynamic programming technique (apply)

Course Content:

UNIT- I - Introduction and overview of C Programming:

Introduction to C programming, Variables, Data types, Constants, Declarations, Operators, Precedence, Associativity, Order of evaluation. Input and output statements; Control Statements, Arrays – Single dimension, Two-dimensional, Multi-dimensional Arrays, Strings. Functions, Categories of functions. Examples Pointers, Pointer arithmetic, Call by value, Pointer Expression, Pointer as function arguments, recursion, passing strings to functions, Call by reference, Functions returning pointers, Pointers to functions, Programming Examples. Structures and Unions.

UNIT- II - Introduction to Data Structures:
Information and its meaning: Abstract Data Types, Sequences as Value Definitions, ADT for Varying length character Strings, pointers and review of Pointers, Dynamic Memory Allocation - definition, malloc, calloc, and realloc, free. Data Structures: Array as an ADT, Arrays as Parameters, String as an ADT.

UNIT- III- The Stack:
Definition and examples, Primitive operations, Example, The stack as an ADT, Representing stacks, Implementing the pop, push operations using function overloading, Examples for infix, postfix, and prefix expressions, Basic definition and Examples. Applications of Stacks: Expression Evaluations, Expression conversion, Recursion as application of stack, Properties of recursivedefinition or algorithm. Binary search, Towers of Hanoi problem.

UNIT- IV- Queues and Linked Lists:
The queue and its sequential representation, the queue as ADT, Basic operations using polymorphism and inheritance, Priority queue, Array implementation of a priority queue. Linked lists, inserting and removing nodes from a list, Linked implementations of stacks, Linked implementation of queues, linked list as a data Structure. Example of list operations.

UNIT- V - Linked Lists and Trees:
Other list structures: Circular lists, Stack as circular lists, doubly linked lists. Application of linked lists: Stacks, Queues, double-ended queues, priority queues. Sorting and Searching: Applications and implementation with function overloading. Tree: Definition and representation, Types of trees, Basic operations on Tree.

Reference Books:

1. Programming in ANSI C, Third Edition, E. Balaguruswamy. 6th Edition (2013).
2. Data Structures Using C and C++ by Aaron.M. Tenenbaum, Yedidyah Langsam and Moshe J.Augustine, PHI, Edition, 2011.
3. Data structures, Algorithms and Applications in C++, S. Sahani, University Press (India) Pvt Ltd,2nd Edition.
4. The complete reference C, Herbert Schildt, Fifth Edition, Tata McGraw Hill.

Course Code:

Course Name: Operating Systems and Linux

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

This course presents to make students have an understanding of the design issues of different aspects of operating systems. To make participants have an in depth understanding of the various OS services for threads, inter-process communication, process synchronization, process and memory management, and file systems, offered as system or library calls in Linux operating system.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Recognize the structure of operating system, interaction of an operating system and application programs.
CO2	Analyze the various programming paradigms viz. multi-process and multi-threaded programming.
CO3	Examine the various resources and memory management techniques.
CO4	Distinguish the different features of real time and mobile operating systems.
CO5	Identify current issues in system security; demonstrate various factors can influence the overall performance of an operating system.

Course Content:

UNIT-I-Computer and Operating Systems Structure:

Basic Elements, Processor Registers, Instruction Execution, The Memory Hierarchy, Cache Memory, I/O Communication Techniques, Introduction to Operating System, Mainframe Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Clustered Systems, Real - Time Systems, Handheld Systems, Feature Migration, Computing Environments.

System Structures: System Components, Operating – System Services, System Calls, System Programs, System Structure, Virtual Machines, System Design and Implementation, System Generation.

UNIT-II-Process Management and Mutual Execution:

Process, Process States, Process Description, Process Control, Execution of the Operating System, Security Issues, Processes and Threads, Symmetric Multiprocessing(SMP), Micro kernels, CPU Scheduler and Scheduling. Principles of Concurrency, Mutual Exclusion: Hardware Support, Semaphores, Monitors, Message Passing, Readers/Writes Problem.

UNIT-III- Deadlock and Memory Management:

Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, An Integrated Deadlock Strategy, Dining Philosophers Problem Memory Management: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Process

Creation, Page Replacement, Allocation of Frames, Thrashing

UNIT-IV-File System and Secondary Storage:

File Concept, Access Methods, Directory Structure, File System Mounting, File Sharing, Protection, File – System Structure, File – System Implementation, Directory Implementation, Allocation Methods, Free–Space Management**,** Disk Structure, Disk Scheduling, Disk Management.

UNIT-V- Computer Security and Shell Programming:

The Security Problem, User Authentication, Program Threats, System Threats. Linux System Linux history, Design Principles, Kernel modules, Process, management, scheduling, Memory management, File systems, Input and output, Inter-process communications.

Shell Programming: Introduction, Shell as a Programming Language, Types of shells, shell syntax, pipes and Redirection, Environment variables, working with files.

Reference Books:

1 Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.

2 William Stallings, "Operating System Internals and Design Principles" Pearson, 6thedition, 2012

3 Dhananjay M. Dhamdhere, "Operating Systems – A Concept – Based Approach",TataMcGraw – Hill, 3rd Edition, 2012

4 Harvey M Deital: Operating systems, 3rEdition, Pearson Education, 1990.

5 Chakraborty , "Operating Systems" Jaico Publishing House, 2011

6 Daniel P.Bovet & Macro Cesati(2006). Understanding Linux Kernel (3rd edition), O'reily Series

7 MichaelBeck, Harald Bohme, Robert Magnus, DirkVerwoner.(2002).LinuxKernelProgramming Pearson Education Ltd.

Course Code:

Course Name: Computer Networks

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
3 Hrs./Week	-	3	30	-	-	70	100

Course Description:

This course aims to enable learners

1. To familiarize the fundamental concepts of computer networking models.
2. To understand different components of computer networks and various protocols with their applications
3. To understand basics of Cryptography and methods to secure a message over insecure channel by various classical encryption techniques.
4. To learn about modern cryptographic techniques such as DES, AES and Secure Hash Algorithms(SHA).

Course Outcome:

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

COURSE OUTCOMES	
CO1	Explore the basic concepts, principles, and techniques of data Communication along with the layers of OSI and TCP/IP model.
CO2	Discuss various error detection and correction techniques in data link layer and analyze the performance of network layer
CO3	Demonstrate the services of transport layer and the basic concepts of Application-Layer Paradigms.
CO4	Explore the fundamental concepts of Cryptography by applying various Encryption techniques.
CO5	Analyze various symmetric ciphers and data integrity algorithms.

Course Content:

UNIT-I-Basics of Data Communications and Physical Layer:

Data Communications: Components, Data Representation, Data Flow, Networks; Network Criteria, Physical Structures, Network Types: LAN, WAN, Switching, Network Models: Protocol Layering: Principles of Protocol Layering, Logical Connections, TCP/IP Protocol Suite: Layered Architecture, Layers in the TCP/IP Protocol Suite,, Addressing, Multiplexing and De- multiplexing, The OSI Model; OSI versus TCP/IP, Lack of OSI Model's Success, Introduction to Physical Layer, Data and Signals, Periodic Analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance, Switching: Circuit-Switched Networks, Packet Switching

UNIT-II-Data Link Layer and Network Layer:

Introduction to Data-Link Layer, Link-Layer Addressing: Address Resolution Protocol (ARP), Error Detection and Correction: Introduction, Types of Errors, Redundancy, Detection versus Correction, Coding, Block coding: Error Detection, Cyclic Code: Cyclic Redundancy Check, Introduction to Network Layer, Network-Layer Services: Packetizing, Routing and Forwarding, Packet Switching; Datagram Approach, Virtual-Circuit Approach, Network Layer.

UNIT-III-Transport layer and Application layer:

Introduction to Transport-Layer: Transport- Layer Services: Transport-Layer Protocols. Introduction to Application Layer, Services, Application-Layer Paradigms, World Wide Web and HTTP: FTP: Two Connections, Control Connection, Data Connection, Security for FTP, E-Mail: Architecture, Web-BasedMail, Domain Name System (DNS): Name Space, DNS in the Internet, Resolution.

Self-Learning Component: Caching, Resource Records, DNS Messages, Registrars, DDNS, Security of DNS.

UNIT-IV-Introduction to Cryptography:

OSI Security Architecture, Security attacks, Security Services, Security Mechanisms, Model for Network Security, Standards, Classical Encryption techniques, Traditional block cipher structure, DES, DES Example.

UNIT-V-AES, Asymmetric Ciphers and Data Integrity Algorithms:

AES Structure, AES transformation functions, AES key expansion, AES Example, AES Example, Principles of Public-key Cryptosystems, RSA Algorithms, Diffie-Hellman Key exchange, Applications ofCryptographic hash functions, two simple hash functions, Hash functions based on Cipher block chaining, Secure hash Algorithm(SHA), Message Authentication Requirement, Message authentication Functions, Message authentication codes.

Reference Books:

1. B. A. Forouzan, "Data Communications and Networking", 5th Edition, McGraw Hill Education (India) Private Limited, 2016.
2. William Stallings, "Cryptography and Network Security", 7th Edition, Pearson, 2018
3. William Stallings, Data and Computer Communications, 10th Edition, Pearson, 2015.
4. Andrew S. Tanenbaum, Computer Networks, Fourth Edition, PHI, 2014.
5. Behrouz A. Forouzan, "Cryptography and Network Security", Tata McGraw-Hill Publishing, 2010.

Course Code:

Course Name: **Computer Organization & Architecture**

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

- Provide a clear understanding of the basic structure and operation of a digital computer.
- Study different number system representation and conversion from one number system to another and Boolean algebra.
- Learn the working of flip-flops, logical gates, multiplexers, and adders.
- Study the memory system cache memories and virtual memory.
- Explain the different ways of communicating with I/O devices and standard I/O interfaces.

Course Outcome:

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

COURSE OUTCOMES	
CO1	Understand the basic of Digital Systems
CO2	Realize the concept of Computer System Organization
CO3	Apply the concepts of Input/output Organization and Memory System
CO4	Analyze the Performance of Memory System and Memory Management
CO5	Analyze the performance bench marks

Course Content:

UNIT- I -Binary Systems and Combinational Logic:

Digital Computers and Digital Systems, Binary Numbers, Number Base Conversion, Octal and Hexadecimal Numbers, subtraction using r's and r-1 complements, Binary Code, Binary Storage and Registers, Binary Logic, Integrated Circuits. Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations, Digital Logic Gates, The map Method, Two – and Three – Variable Maps, Four – Variables Map .

UNIT- II-Arithmetic Circuits and Sequential Logic:

NAND and NOR Implementation, Other Two- Level Implementations, Don't Care Conditions. Introduction, Adders, Subtractors, Binary Parallel Adder, Decimal Adder, Magnitude Comparator, Decoders, Multiplexers, BOOTH algorithm for signed numbers with example. Sequential Logic: Introduction, different types of Flip – Flops, Triggering of Flip- Flops.

UNIT- III -Assembly Language and Input /Output Organization:

Computer Types, Functional Units, Basic Operational Concepts, Bus structure, Software, Performance,

Multiprocessing and Multi computers, Machine Instruction: Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes.

Basics of Assembly Language Programme, Examples from Assembly Language Programming. Accessing I/O Devices, Interrupts, Enabling & Disabling Interrupts, Handling Multiple devices, Controling I/P O/P device behavior, Exceptions, DMA, Buses.

UNIT- IV-The Memory System:

Basic Concepts, Semiconductor RAM Memories, Internal organization of memory chips, static memories, dynamic RAM, Synchronous D-RAM, Structure of larger Memories. Read – Only Memories, Speed, Size, and Cost, Cache Memories, Virtual Memories, Memory Management Requirements.

UNIT- V-Performance Evaluation:

Performance evaluation-SPEC marks, Transaction Processing benchmarks.

Reference Books:

1. M.Morris Mano, "Digital Logic and Computer Design", Pearson, 2012.
2. CarlHamacher, Zvonko Vranesic Safwat Zaky, "Computer Organization", 5thedition, TataMcGraw-Hill, 2011
3. JohnP.Hayes, "Computer Architecture and Organization", Tata McGraw-Hill, Edition, 2012.
4. Soumitrs Kumar Mandal,"Digital Electronics Principles and Applications", Tata McGraw-Hill, 2010
5. Hamacher , " Computer Organization" , McGraw-Hill Education

Course Code:

Course Name: Data Structure and Problem Solving Using C Lab

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
--	2 Hrs./Week	1	-	20	-	30	50

Course Outcome:

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

CO1	Design and analyze programming problem statements.
CO2	Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
CO3	Apply mathematical abstraction to solve problems.
CO4	Demonstrate various methods of organizing large amounts of data.
CO5	Analyze algorithms and to determine algorithm correctness and time efficiency class.

LIST OF EXPERIMENTS COVERED:

PROGRAMS ON C.

1. Programs to learn and explore C data types, looping and decision making structures. {mean, median, lcm, gcd, min max}
2. Calculate the salary of an employee given his basic pay, HRA = 10% of basic pay, TA=5% of his basic pay and deductions IT = 2.5% of his basic pay.
3. Solve quadratic equations to find the roots of the equation.
4. Programs to implement arrays and structures.{Ex: Students marks calculation, matrix operations}
5. Calculate the average marks of the student test marks and display the result using structure.
6. Programs to implement dynamic memory allocation: malloc, calloc, realloc and free.

STACK

7. Write a C program to evaluate the validity of an expression
8. Write a C program to evaluate a postfix expression.
9. Write a C program to convert an expression from infix to postfix.
10. Write a C program to implement multiple stack of integers.

QUEUES

11. Write a C program to perform basic operations on queue of integers, the program should provide the appropriate message to handle all concerned conditions
12. Write a C program to perform basic operations on list of students information stored in circular queue.Let student information include regno, course title, year of stud
13. Write a C program to implement dual queue.

14. LINKED LIST

15. Write a C program to implement stack operations using linked list.
16. Write a C program to implement queue operations using linked list.
17. Write a C program to create the students mark list based on the rank. Let the student record

contain student-id, name, total marks.

18. Write a C program to perform operations.

a. Creation of list.

b. Insertion of new element [At Front, from rear, based on the position]

c. Deletion of a node [At Front, from rear, based on the position]

d. Display the list.

e. Replace the content of one element by another element.

f. Swap two nodes

19. Write a C program to perform the following operations on doubly linked list.

a. Creation of list by :

Insertion [At beginning, At end, In between] Deletion [At beginning, At end, In between]

b. Display all the nodes.

c. Swap two nodes based on specific criteria.

TREES

20. Write a C program to perform / implement the binary tree using array and hence perform the following

a. To print the left and right child of specified node

b. To print all the ancestors of a specified node

c. To print all the node in a specific level

d. To print only the leaf node

21. Write a C program to perform / implement the binary tree using linked list and hence perform the following

a. To print the left and right child of specified node

b. To print all the ancestors of a specified node

c. To print all the node in a specific level

d. To print only the leaf node

22. Write a C program with recursive routines to traverse the binary tree in all possible orders

a. Create a tree

b. Pre-Order traversal

c. In-Order traversal

d. Post-Order traversal

1. Write a C program to construct a heap of n integers and hence sort them using heap sort Algorithm

2. Implement the search techniques

a. Linear Search

b. Binary Search

Reference Books:

1	Programming in ANSI C, Third Edition, E. Balaguruswamy. 6th Edition (2013).
2	Data Structures Using C and C++ by Aaron.M. Tenenbaum, Yedidyah Langsam and Moshe,J. Augustine , PHI, Edition, 2011.
3	Data structures, Algorithms and Applications in C++, S. Sahani, University Press (India)Pvt Ltd, 2nd Edition.
4	The complete reference C, Herbert Schildt, Fifth Edition, Tata McGraw Hil

Course Code:

Course Name: **Computer Networks Lab**

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
--	2 Hrs./Week	1	-	20	-	30	50

Course Outcome:

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

CO1	To understand the working principle of various communication protocols.
CO2	To analyze the various routing algorithms
CO3	To know the concept of data transfer between nodes

LIST OF EXPERIMENTS COVERED:

PART A

1. Write a program for distance vector algorithm to find suitable path for transmission
2. Using TCP/IP sockets, write a client-server program to make the client send the file name and to make the server send back the contents of the requested file if present.
3. Implement the above program using message queues or FIFOs as IPC channels.
4. Write a program for Hamming code generation for error detection and correction.
5. Write a program for congestion control using leaky bucket algorithm.

PART B

Introduction to NS2

1. Simulate a three nodes point — to — point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.
2. Simulate the network with five nodes n0, n1, n2, n3, n4, forming a star topology. Thenode n4 is at the center. Node n0 is a TCP source, which transmits packets to node n3(a TCP sink) through the node n4. Node n1 is another traffic source, and sends UDP packets to node n2 through n4. The duration of the simulation time is 10 seconds
3. Simulate to study transmission of packets over Ethernet LAN and determine the number of packets drop destination
4. Write a TCL Script to simulate working of multicasting routing protocol and analyze the throughput of the network
5. Simulate the different types of internet traffic such as FTP and TELNET over a wired network and analyze the packet drop and packet delivery ratio in the network.
6. Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion
7. Write a TCL script to simulate the following scenario with ns2 simulator. Consider sixnodes, (as shown in the figure below) moving within a flat topology of 700m x 700m. The initial positions of nodes are 0 (150,300),1 (300,500),2 (500,500),3 (300,100),4(500,100) and 5(650,300) respectively. A TCP connection is initiated between node 0 (source) and node 5 (destination) through node 3 and node 4 i.e the route is 0-3-4-5. At time t = 3 seconds the FTP application runs over it. After time t=4.0 sec, node 3 (300,100) moves towards node 1 (300,500) with a speed of 5.0m/sec and after some time the path break, then the data transmit with a new path via node 1 and node 2 i.e the new route 0-1-2-5. The simulation lasts for 60 secs. In the above said case both the route has equal cost. Use DSR as the routing protocol and the IEEE 802.11 MAC[29] protocol

SOFT CORE SUBJECTS

Course Code:

Course Name: Mathematical Foundations for Computer Science

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

This course will discuss fundamental concepts and tools in discrete mathematics with emphasis on their applications to computer science. Example topics include logic and Boolean circuits; sets, functions, relations, databases, and finite automata; deterministic algorithms and randomized algorithms; analysis techniques based on counting methods and recurrence equations; trees and more general graphs.

Course Outcome:

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

CO1	Implement statistical measures and explore its applications
CO2	Analysis of computational errors and design of algorithms to solve a set of linear equations.
CO3	Applying the concepts of vector and linear functions in real time applications.
CO4	Apply the notion of relations on finite structures, like strings and analyze algorithms using the concept of functions.
CO5	Explore the properties of Graph theory and its applications in computer science.

Course Content:

UNIT- I-Statistics:

Univariate data – different measures of location, dispersion, relative dispersion, skewness and kurtosis, Moments, Measures based on them – comparison with moment measures, Correlation and Regression Analysis.

UNIT- II-Number Systems and Vector & Matrix Algebra:

Errors in Numerical Computations, Types of Errors, Analysis and Estimation of Errors, Vector Algebra: Vector spaces with real field, Basis and dimension of a vector space, Orthogonal vectors, Properties of Matrices and Determinants: Matrix Operations, Elementary Matrices, Inverse Matrix, Diagonal Matrix, Symmetric Matrix, and Determinant Matrix.

UNIT- III-Linear Algebraic Systems:

Numerical methods for Linear Systems, Direct Methods for Linear Systems: Cramer's Rule, Gauss Elimination Method, Gauss Jordan Elimination Method, Pivoting Strategies, Gauss- Jordon Method, LU Decomposition Method, Tridiagonal Systems of Linear Equations, Iterative Methods for Solving Linear Systems, Jacobis Iteration Method, Gauss-Seidel Iterative Method, Convergence Criteria, Eigen Values and Eigen Vectors.

UNIT- IV-Relations and Functions:

Cartesian products and Relations, Properties of Relations, Functions: Plain and One-to-One, Onto Functions: Stirling Numbers and the Second Kind, Special functions, The Pigeon-hole principle, Function composition and inverse functions.

UNIT- V-Graph Theory:

Definitions and Examples, Subgraphs, Complements, and Graph Isomorphism, Vertex Degree: Euler Trails and Circuits, Planar Graphs, Hamilton Paths and Cycles, Graph Coloring and Chromatic Polynomials.

Reference Books:

1 Sant Sharan Mishra, "Computer Oriented Numerical and Statistical Methods", PHI Learning Private Limited, 2013.

2 Rizwan Butt, "Introduction to Numerical Analysis Using Matlab", Infinity Science Press LLC, 2008

3 Ralph P Grimaldi, B.V.Ramana, "Discrete & Combinatorial Mathematics, An Applied Introduction" 5th Edition, Pearson Education, 2009.

4 D.S. Chandrasekharaiah, Discrete Mathematical Structures, 4th Edition, PRISM Pvt. Ltd. 2012.

5 Bondy and U.S.R.Murty: Graph Theory and Applications (Freely downloadable from Bondy's website; Google-Bondy)

6 S. Kumarsean, "Linear Algebra A geometric approach", Prentice Hall of India Private Limited, 2001

7 Kenneth H Rosen, "Discrete Mathematics & its Applications" 7th edition, McGraw- Hill, 2010.

Course Code:

Course Name: Business Systems

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

This subject gives overview about business and information system for business. It presents various information technology used for effective management of business activities.

Course Outcome:

Upon successful completion of this course, students will have the ability to,

COURSE OUTCOMES	
CO1	Understand the concepts of business, and information system for business.
CO2	Adopt of Information Technology in business for effective management of activities.
CO3	Develop Business and Corresponding IT strategies.
CO4	Design appropriate Business / IT solutions as per the business requirement
CO5	Understand the concepts and essential of e-business systems

Course Content:

UNIT- I-Information Systems in Business:

The Fundamental Roles of IS in Business, Trends in Information Systems, The Role of e- Business in Business, Types of Information Systems - Operations Support Systems, Management Support Systems, Other Classifications of Information Systems.

UNIT- II-Information Technology in Business:

Managerial Challenges of Information Technology, Components of Information Systems, Information System Resources, Information System Activities, Recognizing Information Systems.

UNIT- III-Developing Business/IT Strategies:

Planning Fundamentals, Organizational Planning- The Scenario Approach, SWOT Analysis, Business Models and Planning, Business/IT Architecture Planning, Identifying Business/IT Strategies, Business Application Planning, Change Management.

UNIT- IV-Designing Business/IT Solutions:

Introduction, Cross-Functional Enterprise Applications, Enterprise Application Integration,Transaction Processing Systems, Enterprise Collaboration Systems.

UNIT-V-e-Business Systems:

Information Technology in Business, Marketing Systems, Manufacturing Systems, HumanResource Systems, Accounting Systems, Financial Management Systems.

Reference Books:

1 James A. O'Brien and George M. Marakas, Management Information System, TenthEdition, Tata McGraw Hill. 2013.

2 Jawadekar, W.S., "Management Information Systems", Tata McGraw Hill PrivateLimited, New Delhi, 2009.

Course Code:

Course Name: Data Mining and Data Warehousing

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
3 Hrs./Week	2 Hrs./Week	3	15	15	-	70	100

Course Description:

To identify the scope and essentiality of Data Warehousing and Mining. To analyze data, choose relevant models and algorithms for respective applications. To study spatial and web data mining. To develop research interest towards advances in data mining.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Acquire the knowledge of data preprocessing and data quality; modeling and design of data warehouses and algorithms for data mining.
CO2	Be able to design data warehouses and apply acquired knowledge for understanding data and select suitable methods for data analysis.

Course Content:

UNIT -1 – Introduction :

Introduction to data mining and Data Warehousing, Modeling: Data Cube and OLAP, Data Warehouse Implementation, Data Mining – types of data, types of patterns, Data cleaning, Data integration.

UNIT -1I – Data Reduction:

Data Reduction, Wavelet Transforms, Attribute Subset Selection, Histogram, Clustering, Sampling, Data Cube Aggregation Data Transformation: Strategies Overview, Data Transformation by Normalization.

UNIT -1II – Mining Frequent Patterns:

Association Rules: Problem Definition, Frequent Item Set Generation, The APRIORI Principle, Support and Confidence Measures, Association Rule Generation; APRIOIRI Algorithm, The Partition Algorithms, FP-Growth Algorithms, Compact Representation of Frequent Item Set- Maximal Frequent Item Set, Closed Frequent Item Set.

UNIT -1V – Classification:

Problem Definition, General Approaches to solving a classification problem, Evaluation of Classifiers , Classification techniques, Decision Trees-Decision tree Construction, Methods for Expressing attribute test conditions, Measures for Selecting the Best Split, Algorithm for Decision tree Induction ;Naive-Bayes Classifier, Bayesian Belief Networks; K- Nearest neighbor classification-Algorithm and Characteristics.

Prediction: Accuracy and Error measures, Evaluating the accuracy of classifier or a predictor, Ensemble methods

UNIT -V – Cluster Analysis:

Cluster Analysis: Requirement for Cluster Analysis, clustering methods Data Mining Applications & Trends: Mining Sequence Data; Time Series, Symbolic, Statistical Data Mining, Visual Data Mining, Data Mining Applications.

Reference Books:

1 Jiawei Micheline Kamber, 'Data Mining Concepts and Techniques', Morgan Kauf Mann Publishers.

2 George M. Marakas, 'Modern Data Warehousing, Mining and Visualization', Pearson Education, 2003.

3 W.H. Inmon, 'Building the Data Warehouse', Wiley dreamtech, 3rd Edition.

4 Mastering Data Mining – Michael J.A. Berry & Gordon S. Linoff (Wiley Pub.).

5 Data Warehousing (Pearson Ed.) – Sam Anahory & Dennis Murray.

Course Name: Data Mining and Data Warehousing (Practical Component)

Course Description:

1. Learn how to build a data warehouse and query it (using open source tools like Pentaho Data Integration Tool, Pentaho Business Analytics).
2. Learn to perform data mining tasks using a data mining toolkit (such as opensource WEKA).
3. Understand the data sets and data preprocessing.
4. Demonstrate the working of algorithms for data mining tasks such association rule mining,classification, clustering and regression.
5. Exercise the data mining techniques with varied input values for different parameters.
6. To obtain Practical Experience Working with all real datasets.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Ability to add mining algorithms as a component to the existing tools
CO2	Demonstrate the classification, clustering and etc. in large data sets.
CO3	Ability to apply mining techniques for realistic data.

Sl.No.	List of Experiments
1	List all the categorical (or nominal) attributes and the real-valued attributes separately.
2	What attributes do you think might be crucial in making the credit assessment? Come up with some simple rules in plain English using your selected attributes.
3	One type of model that you can create is a Decision Tree -train a Decision Tree using the complete dataset as the training data. Report the model obtained after training.
4	Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? (This is also called testing on the training set) Why do you think you cannot get 100 % training accuracy?
5	Is testing on the training set as you did above a good idea? Why or Why not?
6	One approach for solving the problem encountered in the previous question is using cross-validation? Describe what is cross -validation briefly. Train a Decision Tree again using cross - validation and report your results. Does your accuracy increase/decrease? Why?
7	Check to see if the data shows a bias against "foreign workers" (attribute 20), or "personal -status" (attribute 9). One way to do this (perhaps rather simple minded) is to

	remove these attributes from the dataset and see if the decision tree created in those cases is significantly different from the full dataset case which you have already done. To remove an attribute you can use the preprocess tab in Weka's GUI Explorer. Did removing these attributes have any significant effect? Discuss.
8	Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones.
9	Sometimes, the cost of rejecting an applicant who actually has a good credit (case 1) might be higher than accepting an applicant who has bad credit (case 2). Instead of counting the misclassifications equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. You can do this by using a cost matrix in Weka. Train your Decision Tree again and report the Decision Tree and cross -validation results. Are they significantly different from results obtained in problem 6 (using equal cost)?
10	Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model?
11	You can make your Decision Trees simpler by pruning the nodes. one approach is to use Reduced Error Pruning -Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross -validation (you can do this in Weka) and report the Decision Tree you obtain? Also, report your accuracy using the pruned model. Does your accuracy increase?
12	(Extra Credit): How can you convert a Decision Trees into "if –then -else rules". Make up your own small Decision Tree consisting of 2 - 3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules -one such classifier in Weka is rules. PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one! Can you predict what attribute that might be in this dataset? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR.

Course Code:

Course Name: Probability and Statistics

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

To equip the students with a working knowledge of probability, statistics, and modeling in the presence of uncertainties. To understand the concept of hypothesis and significance tests. To help the students to develop an intuition and an interest for random phenomena and to introduce both theoretical issues and applications that may be useful in real life.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Distinguish between quantitative and categorical data
CO2	Apply different statistical measures on data
CO3	Identify, formulate and solve problems
CO4	Classify different types of Probability and their fundamental applications

Course Content:

UNIT-I-Measures of Central Tendency &Measures of Dispersion:
Frequency Distribution, Histogram, Stem and leaf diagram,ogives, Frequency Polygon, Mean, Median, Mode, Range, Quartile Deviation, Mean Deviation, Box whisker plot, Standard Deviation, Coefficient of Variation.

UNIT-II-Skewness, Correlation &Regression:
Karl Pearson's coefficient of Skewness, Bowley's coefficient of Skewness, Scatter Diagram, Karl Pearson's coefficient of correlation, Spearman's rank correlation coefficient , Linear Regression and Estimation, coefficients of regression.

UNIT-III-Theory of Attributes:
Classes and Class Frequencies, Consistency of Data, Independence of Attributes, Association of Attributes
Testing of Hypothesis: Hypothesis, Type I and Type II errors. Tests of significance – Student's t-test:Single Mean, Difference of means, paired t-test, Chi-Square test:Test of Goodness of Fit, Independence Test.

UNIT-IV-Introduction toProbability:
Random experiment, Sample space, Events, Axiomatic Probability, Algebra of events
Conditional Probability: Conditional Probability, Multiplication theorem of Probability, Independent events, Baye's Theorem.

UNIT-V-Randomvariables:
Discrete random variable, Continuous random variable, Two-dimensional random variable, Joint probabilitydistribution, Stochastic independence
MathematicalExpectation: Expected value of a random variable, Expected value of a function of a random variable, Properties of Expectation an Variance, Covariance.

Reference Books:

1 Fundamentals of Mathematical Statistics – 1st Edition S.C.Gupta, V.K.Kapoor , S Chand
2 Introduction to Probability & Statistics – 4th Edition J.Susan Milton, Jesse C. Arnold Tata McGraw Hill
3 Fundamentals of Statistics : 7th edition S C Gupta, Himalaya Publishing house
4 Probability and Statistics with Reliability, Queuing, And Computer Science Applications (English) 1st Edition: Kishore Trivedi, PHI
5 Schaum's Outlines Probability, Random Variables & Random Process 3rd Edition Tata McGraw Hill
6 Probability & Statistics for Engineers: Dr J Ravichandran, Wiley
7 Statistics for Business and Economics: Dr Seema Sharma, Wiley
8 Applied Business Statistics 7th Edition Ken Black, Wiley

SEMESTER 2
HARD CORE SUBJECTS

Course Code:

Course Name: Java Programming

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
4 Hrs./Week	-	3	30	-	-	70	100

Course Des cription:

This course of study builds on the skills gained by students in Java Fundamentals or Java Foundations to help advance Java programming skills. Students will design object-oriented applications with Java and will create Java programs using hands-on, engaging activities.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Demonstrate and implement programs using components and constructs of a Java language
CO2	Identify classes, objects, members of a class and use packages and interfaces appropriately.
CO3	Demonstrate for Java program formulate thread, synchronization and exception handling concepts.
CO4	Use the concept of string, event handling, simple data structures like arrays and members of classes of Java API in application development.
CO5	Design and develop Java based UI and Networking applications using applets, swing components and networking concepts.

Course Content:

UNIT-I-Java Basics:

History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner classes, exploring string class

UNIT-II-OOP Concepts in Java, Packages and Interfaces:

Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance-specialization,specification,construction,extension,limitation,combination,benefitsofinheritance, costs

of inheritance. Member access rules, super uses, using final with inheritance, polymorphism-method overriding, abstract classes, the Object class.

Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring java.io.

UNIT-III-Exception Handling and Multi-Threading:

Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes. String handling, Exploring java.util.

Differences between multi-threading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads; inter thread communication, thread groups, daemon threads. Enumerations, auto boxing, annotations, generics.

UNIT-IV-String and Event Handling:

String fundamentals, String Constructors, Three string related language features, The Length () method, Obtaining the characters within the string, String comparison, using index Of () and last index Of (), changing the case of the characters within the string, String buffer and String builder.

Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes. The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, lists panels –scroll pane, dialogs, menu bar, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

Reference Books:

1 Herbert Schildt. Java - The Complete Reference, Ninth Edition. Oracle Press, McGraw Hill Education (India) Edition-2014.

2 Cay S. Horstmann, Gary Cornell. Core Java, Core Java Volume-1 – Fundamentals, 9th edition, Pearson Education, 2014.

3 Timothy Budd, "Understanding Object-oriented programming with Java", Updated Edition, Pearson Education, 2000.

Course Code:

Course Name: Advanced Database Management Systems

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
3 Hrs./Week	-	3	30	-	-	70	100

Course Description:

This course will enable student to

- Understand the fundamental concepts of Database Management systems.
- Design ER Diagrams, Schema and Relational tables.
- Understand how to develop database Application.

Course Outcome:

Upon successful completion of this course, the student will be able to:

CO1	Demonstrate the fundamentals of data models and conceptualize and depict a database system and Make use of ER diagram in developing ER Model
CO2	To Summarize the SQL and relational database design.
CO3	Illustrate transaction processing, concurrency control techniques and recovery
CO4	Inference the database design in the real world entities.

Course Content:

UNIT- I- Introduction DBMS – Concepts & Architectures:

Database and Need for DBMS, Characteristics of DBMS Database 3-tier schema (ANSI/SPARC) and system architecture of DBMSViews of data- Schemas and instances, Data Independence Centralized, Client-Server system, Transaction servers, Data servers, Cloud based servers Indexing and Hashing - Basic concepts of indexing, ordered index, B+ tree index, B+ treeextensions, Multiple key access, Hashing concepts, types of hashing, Bitmap indices.

UNIT- II- Data Modelling and Relational Database Design and Database Design:

Data Modelling using ER Diagram: Representation of Entities, Attributes, Relationships and their Type, Cardinality, Generalization, Specialization, Aggregation.
Relational data model: Structure of Relational Database Model, Types of keys, Referential Integrity Constraints, Codd's rules, Database Design using E-R, E-R to Relational
Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms.

UNIT- III- Introduction to SQL:

Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the

Database, Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization. Database programming issues and techniques, Embedded SQL.

UNIT- IV- Transaction and Concurrency control:

Concept of transaction, ACID properties, States of transaction Concurrency control, Problems in concurrency controls Scheduling of transactions, Serializability and testing of serilaizibity Lock-based Protocol and Time stamp-based ordering protocols, Deadlock Handling

Crash Recovery and Backup, Failure classifications, Recovery & Atomicity, Log based recovery, Checkpoint and Shadow Paging in Data recovery, Database backup and types of backups

UNIT- V -Object Oriented Databases & Applications:

Overview of Object- Oriented Database concepts & characteristicsDatabase design for OODBMS – Objects, OIDs and reference typeSpatial data and Spatial indexing (Any two techniques)

Mobile Database: Need, Structure, Features, Limitations and Applications Temporal databases, temporal aspects valid time, transaction time or decision timeMultimedia Database: Architecture, Type and Characteristics.

NO-SQL Database

Introduction, Types of NOSQL, Need of NoSQL databases, Use Cases

Reference Books:

1. Introduction to database systems C.J. Date, Pearson.
2. Fundamentals of Database Systems by Elmasri Navathe
3. Principles of Database Management James Martin, PHI
4. Database System Concepts by Abraham Silberschatz, H. Korth, Sudarshan
5. Database Management System by Raghu Ramakrishnan / Johannes Gherke
6. Database Management System (DBMS) A Practical Approach. By Rajiv Chopra
7. Database system practical approach to design, implementation & management by Connoly &Begg,
8. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence Martin Fowler

Course Code:

Course Name: Java Programming Lab

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
--	2 Hrs./Week	1	-	30	-	70	100

Course Outcome:

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

COURSE OUTCOMES	
CO1	Demonstrate and implement programs using components and constructs of a Javalanguage
CO2	Identify classes, objects, members of a class and exhibit use packages and interfaces appropriately.
CO3	Demonstrate for Java program for multithread, synchronization and exception handlingconcepts.
CO4	Use the concept of string, event handling, simple data structures like arrays and membersof classes of Java API in application development
CO5	Design and develop Java based UI and Networking applications using applets, swingcomponents and networking concepts.

LIST OF PROGRAMS TO BE COVERED:

1. Display Hello world
2. Check entered number is ODD or EVEN
3. Find factorial of number
4. Find the sum of the digits of a given number
5. Swap two numbers without using a temporary variable
6. Accept a name and display the name with greeting message using Class.
7. Generate a salary for an employee using class, object, constructors, methods and access control. Different parameters to be considered are Emp_No, Emp_Name, Age, Basic, DA,HRA, CA, PT, IT.
8. Generate a sales report for a sales executive using class, object, constructors, methods and access control. Different parameters to be considered are Emp_No, Emp_Name, Sales_Q1,Sales_Q2, Sales_Q3, Sales_Q4.
9. Demonstrate Constructor Overloading and Method Overloading.
10. Implement Inner class and demonstrate its Access protection.
11. Write a program in Java for String handling which performs the following:
    a. Checks the capacity of String Buffer objects.
    b. Reverses the contents of a string given on console and converts the resultant stringin upper case.
    c. Reads a string from console and appends it to the resultant string of ii.
12. Demonstrate Inheritance.
13. Simple Program on Java for the implementation of Multiple inheritance using
    a. interfaces to calculate the area of a rectangle and triangle.

14. Write a JAVA program which has
    a. A Class called Account that creates account with 500Rs minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount andalso throws Less Balance Exception if an account holder tries to withdraw money which makes the balance become less than 500Rs.
    b. A Class called Less Balance Exception which returns the statement that sayswithdraw amount (Rs) is not valid.
    c. A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a Less Balance Exceptiontake appropriate action for the same.
15. Write a JAVA program using Synchronized Threads, which demonstrates ProducerConsumer concept.
16. Write a JAVA program to implement a Queue using user defined Exception Handling (alsomake use of throw, throws.).
17. Complete the following:
    a. Create a package named shape.
    b. Create some classes in the package representing some common shapes like Square,
    c. Triangle and Circle. Import and compile these classes in other program.
18. Write a JAVA Program
    a. Create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method is Workday( ) to the Day of Week class that returns true if the value on which it is called is MONDAY through FRIDAY. For example,the call Day Of Week SUNDAY is Workday ( ) returns false.
19. Write a JAVA program which has
    a. A Interface class for Stack Operations
    b. A Class that implements the Stack Interface and creates a fixed length Stack.
    c. A Class that implements the Stack Interface and creates a Dynamic length Stack.
    d. A Class that uses both the above Stacks through Interface reference and does theStack operations that demonstrates the runtime binding.
20. Print a chessboard pattern.
21. Write a JAVA Program which uses File Input Stream / File Output Stream Classes.
22. Demonstrate utilities of Linked List Class.
23. Write a JAVA applet program, which handles keyboard event.
24. Write a JAVA Swing program, to design a form.
25. Create a simple Student Registration application using Swings, JDBC and MySQL.
26. Write a JAVA program which uses Datagram Socket for Client Server Communication.

Reference Books:

1 Herbert Schildt. Java - The Complete Reference, Ninth Edition. Oracle Press, McGraw HillEducation (India) Edition- 2014.

2 Cay S. Horstmann, Gary Cornell. Core Java, Core Java Volume-1 – Fundamentals, 9th edition, Pearson Education, 2014.

3 Timothy Budd, “Understanding Object-oriented programming with Java”, Updated Edition, Pearson Education, 2000.

Course Code:

Course Name: Advanced Database Management Systems Lab

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
--	2 Hrs./Week	1	-	30	-	70	100

Course Outcome:

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

COURSE OUTCOMES	
CO1	Implementation of data models, modeling notations, SQL queries.
CO2	Design and implement a database schema for given problem.
CO3	Design and build a simple database system considering the different concepts of RDBMS.

LIST OF PROGRAMS TO BE COVERED:

Introduction Lab

Introduction Lab :

Basic Concepts of DBMS will be covered Like:

- Types of DBMS
- DBMS Languages
- SQL-Structured Query Language
- Basic Instructions given with respect to Lab
- Installation Procedure given to students with respect to MySQL Software

CREATE, USE, INSERT, DESC and SELECT commands willbe explained.

1	**Database Number -1 : Banking Database** Consider the following relations: BRANCH (branch name: varchar , City : Varchar , Asset : Varchar) ACCOUNT (account Number : Varchar, Branch name : Varchar, balance: float) CUSTOMER (customer name : Varchar, Street : Varchar , City : Varchar) DEPOSITOR (customer name : varchar, account number : varchar) LOAN (loan number : varchar, branch name : varchar, amount : integer)BORROWER (customer name : varchar, loan number : varchar) • The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. • Primary Key and Foreign Key relationships will be explained through demo.
	Database Number -1 : Banking Database (Continuation) Queries : • List the loan number from loan having amount 10000 with a specific branch name. • List the loan number with amount between 1000 and 10000. • List the cname with substring. • List the number of tuples in customer. • List customer name, loan num and amount with specific branch name. • Various Aggregate functions will be used to retrieve the data from the aboveconstructed databases. • Various Comparison operators will be used to retrieve the data from the aboveconstructed databases • Various Logical operators will be used to retrieve the data from the aboveconstructed databases
2	**Database Number -2 : Accessories Database** Consider the following relations: PRODUCT (maker: varchar, model : varchar, type: varchar) PC (model : varchar , speed : varchar, ram : varchar, hdd: varchar , removable disk : int ,price : int) LAPTOP (model : varchar , speed : varchar, ram : varchar, hdd: varchar , screen: varchar, price : int) PRINTER (model: varchar, color : blob , type : varchar, price : int) • The tables will be created using CREATE, DESC and tuples will be inserted using
	INSERT commands and display using SELECT Commands. • Primary Key and Foreign Key relationships will be explained through demo. • Find the model,speed,RAM,HDD for all pc's, whose price is under 35000. • Rename the speed column to mhz and HDD column to GB in pc. • Find all manufactures of laptop. • Find all the tuples in printer for color. • Various Aggregate functions will be used to retrieve the data from the aboveconstructed databases. • Various Comparison operators will be used to retrieve the data from the aboveconstructed databases • Various Logical operators will be used to retrieve the data from the aboveconstructed databases.

3	**Database Number -3 : Order Processing Database** Consider the following relations: CUSTOMER (Cust id : varchar, cust name : varchar, city : varchar) ORDER (Order num : varchar, order date : date, cust id : varchar , order amount : int) ITEM (item id : varchar , unit price : int) ORDERITEM (order num: varchar, item id: varchar, quantity : int) WAREHOUSE (warehouse id : varchar, city: varchar) SHIPMENT (order num: varchar, warehouse id : varchar, shipdate : date) • The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. • Primary Key and Foreign Key relationships will be explained through demo. • Produce a list in customer name, number of orders, average order amount where the middle column is total number of order by the customers and the last column is average order amount for third column. • List the order number for order that were shipped from all the ware houses that the company asked in specific city. • Various Aggregate functions will be used to retrieve the data from the above constructed databases. • Various Comparison operators will be used to retrieve the data from the above constructed databases • Various Logical operators will be used to retrieve the data from the above constructed databases.
4	**Database Number -4 : Supplier Database** Consider the following relations: SUPPLIER (Supplier id : varchar, name : varchar , address : varchar) PART (part id : varchar, partname : varchar, color : tinyblob) CATALOG (supplier id : varchar, part id : varchar, cost : int) • The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. • Primary Key and Foreign Key relationships will be explained through demo. • Find the names of suppliers who supply only red parts. • Find supplierid of suppliers who supply red and green parts. • Find the supplierid of supplier who supplies some red part OR whose address is'Mysuru'. • Find the suppplierid of suppliers who supply some red and some green parts. • Find the supplierid of suppliers who supply every parts.

5	**Database Number -5 : University Database** Consider the following relations: CLASSROOM (building varchar(30), roomno varchar(30) primary key, capacity int) DEPARTMENT (deptname varchar(30) primary key,building varchar(30),budget decimal(10,2) COURSE(courseid varchar(30) primary key, title varchar(30),deptname varchar(30),credits int) INSTRUCTTOR(iid varchar(30) primary key, name varchar(30), deptname varchar(30),salary int) STUDENT(stdid varchar(30) primary key,name varchar(30),deptname varchar(30),totalcredit int) TEACHES(iid varchar(30),courseid varchar(30), secid varchar(30),sem varchar(30), year varchar(30)) SECTION (courseid varchar(30),secid varchar(30) primary key,sem varchar(30), year varchar(30),building varchar(30),roomno varchar(30),timeid varchar(30)) TIMESLOT(timeid varchar(30) primary key,day varchar(30), starttime time, endtime time); PREREQUIST (courseid varchar(30),preid varchar(30)) ADVISOR (stdid varchar(30), iid varchar(30)) TEXT(iid varchar(30), courseid varchar(30),secid varchar(30),sem varchar(30), year varchar(30), grade varchar(30)) • The tables will be created using CREATE, DESC and tuples will be inserted usingINSERT commands and displayed using SELECT Commands. • Primary Key and Foreign Key relationships will be explained through demo.
	Database Number -5 : University Database (Queries) Continuation • RETREIVE THE NAMES OF THE INSTRUCTORS AND ORDER THEM BY THE DEPARTMENTAL NAME. • RETERIVE WHOSE SALARY IS BETWEEN 10000-20000 • FIND THE AVERAGE SALARY OF INSTRUCTOR FOR A PARTICULAR DEPARTMENT. • FIND THE MINIMUM SALARY OF INSTRUCTOR FOR A PARTICUALR DEPARTMENT • FIND THE AVERAGE SALARY OF THE INSTRUCTOR GROUP BY THEM DEPARTMENTWISES WHERE AVERAGE SALARY SHOULD BE GREATER THEN 50000 • DISPLAY THE NAME OF THE INSTRUCTOR WHO TAKES CLASS IN BUILDING-GOLDENJUBLEE BLOCK. • RETERIVE THE TOTAL NUMBER OF STUDENTS IN EACH DEPARTMENT SECTIONWISE.

	Database Number -5 : University Database (Queries) Continuation • DISPLAY NAME,SALARY FROM INSTRUCTOR OF MBA WHOSE SALARY IS BTWEEN50000-70000 • FIND THE NAMES OF ALL STUDENT WHO HAD TAKEN ATLEAST ONE MCA COURSE • FIND THE DEPARTMENT NAME OF ALL INSTRUCTOR. • FIND THE ENROLLMENT OF EACH SECTION THAT WAS OFFERED IN 2015 AND ODD • DELETE ALL COURSE THAT HAVE NEVER BEEN OFFERED(DON'T OCCUR INSECTION) • RETRIEVE NAMES OF ALL THE DEPARTMENT OF THE SPECIFIC INSTRUCTOR • RETRIEVE ALL THE COURSE TITLES OF A SPECIFIC INSTRUCTOR • INCREASE THE SALARY OF EACH INSTRUCTOR BY 25000 AND LIST THERE NAMESAND ID. • RETRIEVE THE NAMES OF ALL INSTRUCTORS WITH THEIR DEPARTMENT ANDDEPARTMENT BUILDING NAME.
	Database Number -5 : University Database (Queries) Continuation • LIST THE NAME OF THE INSTRUCTOR WHICH INCLUDES THE LETTERS 'MA' ANDTHE COURSED OF ALL COURSES THEY TEACH • LIST THE ENTIRE INSTRUCTORS RELATION IN DESCENDING ORDER OF SALARY IFTHE SEVERAL INSTRUCTOR HAS A SAME NAME. • FIND THE TOTAL NUMBER OF CREDITS OFFERED BY EACH DEPARTMENT • FIND THE AVERAGE SALARY OF ALL INSTRUCTORS OF EACH DEPARTMENT • FIND ALL THE COURSE TAUGHT IN ODD 2018 NOT IN EVEN 2018 • FIND THE NAMES OF THE INSTRUCTOR WHOES NAME ARE NEITHER CHARLIE NORDEEPIKA • CREATE A VIEW THAT LISTT ALL COURSE SECTION OFFERED BY MCA DEPT IN EVEN2016 WITH BUILDING AND ROOMNO FOR EACH SECTION • LIST ALL THE NAME OF THE STUEDNT WITH MAX CGPA • SELECT THE COURSE BELONGING TO MBA DEPT THAT RUNS IN ROOMNO INMBAIS DURING 2015 LIKE '%MA';
	Database Number -5 : University Database (Queries) Continuation • FIND THE TOTAL CGPA SCORED BY THE STDUENT OF EACH DEPARTMENT • FIND THE TOTAL NUMBER OF CREDIT OFFERED BY EACH DEPARTMENT • FIND THE ENROLLMENT OF EACH SECTION THAT HAS OFFERED IN ODD 2019 • LIST THE TOTAL NUMBER OF INSTRUCTOR WHO TEACHES IN ODD 2019 • Students will construct ER diagram and SCHEMA for the above data base.

Reference Books:

1 A. Silberschatz, Henry.F.Korth, S.Sudharshan, "Database System Concepts", 7th Edition,2017
2 Raghu Ramakrishnan and J Gehrke," Database Management Systems", 3rd Edition, 2016.
3 C.J.Date, AKannan, S..Swamynathan ,"An Introduction to Database System", 8th Edition,2009.
4 RamezElmasri, Shamkant.B.Navathe, "Database Systems", 7th Edition,2016.

SOFT CORE SUBJECTS

Course Code:

Course Name: Analysis &Design of Algorithms

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
4 Hrs./Week	2 Hrs./Week	4	15	15	-	70	100

Course Description:

Students study techniques for designing algorithms and for analyzing the time and space efficiency of algorithms. The algorithm design techniques include divide-and-conquer, greedy algorithms, dynamic programming, randomized algorithms and parallel algorithms.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Apply object oriented techniques to solve bigger computing problems
CO2	Explore the knowledge of computational complexity, approximation and randomized algorithms.
CO3	Analyze the range of the algorithm and the notion of tractable and intractable problems.
CO4	Design and analyze a wide range of searching and sorting algorithms.
CO5	Implementation of graph and matching algorithms.

COURSE CONTENT:

UNIT- I - Introduction and the fundamentals of the Analysis of Algorithm Efficiency

Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non-recursive algorithms, Examples.

UNIT -II- Brute Force and Divide and Conquer:

Selection Sort and Bubble Sort, Sequential Search and String Matching, Exhaustive Search, Merge- sort, Quick-sort, Binary Search, Binary tree Traversals and related properties.

UNIT-III- Decrease-and-Conquer, Transform-and-Conquer :

Insertion Sort, Depth First search and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects. Presorting, Balanced Search Trees, Heaps and Heap sort, Problem Reduction.

UNIT-IV - Space and Time Trade offs and Dynamic Programming:

Sorting by Counting, Input Enhancement in String Matching, Computing a binomial coefficient, Warshall's and Floyd's Algorithms, The Knapsack Problem and Memory Functions.

UNIT-V - Limitations of Algorithm Power:

Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems.

Coping with Limitations of Algorithm Power

Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem. Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesperson Problem.

Reference Books:

1 Anany Levitin: Introduction to the Design and Analysis of Algorithms, Pearson Education, 2003.

2 Herbert Scheldt: The Complete Reference C++, 6th Edition, Tata McGraw Hill 2013.

3 Coremen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, PHI, 1998.

4 Horowitz E., Sahani S., Rajasekharan S.: Computer Algorithms, Galgotia Publications, 2001.

Course Name: Algorithm Design and Analysis (Practical component)

Course Outcome:

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

COURSE OUTCOMES	
CO1	Demonstrate and implement programs using components and constructs of a Java language
CO2	Identify classes, objects, members of a class and exhibit use packages and interfaces appropriately.
CO3	Demonstrate for Java program for multithread, synchronization and exception handling concepts.
CO4	Use the concept of string, event handling, simple data structures like arrays and membersof classes of Java API in application development
CO5	Design and develop Java based UI and Networking applications using applets, swing components and networking concepts.

LIST OF PROGRAMS TO BE COVERED:

1. Write a program to sort a list of N elements using Selection Sort Technique.
2. Write a program to perform Travelling Salesman Problem
3. Write a program to sort a list of N elements using Selection Sort Technique.
4. Write program to implement Dynamic Programming algorithm for the 0/1 Knapsack problem.
5. Write a program to perform Knapsack Problem using Greedy Solution
6. Write program to implement the DFS and BFS algorithm for a graph.
7. Write a program to find minimum and maximum value in an array using divide and conquer.
8. Write a test program to implement Divide and Conquer Strategy. Eg: Quick sort algorithm for sorting list of integers in ascending order.
9. Write a program to implement Merge sort algorithm for sorting a list of integers in ascending order.
10. Sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of n> 5000, and record the time taken to sort.
11. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n> 5000 and record the time taken to sort.
12. Write C program that accepts the vertices and edges for a graph and stores it as an adjacency matrix.
13. Implement function to print In-Degree, Out-Degree and to display that adjacency matrix.
14. Write program to implement backtracking algorithm for solving problems like N queens.
15. Write a program to implement the backtracking algorithm for the sum of subsets problem.
16. Write program to implement greedy algorithm for job sequencing with deadlines.
17. Write program to implement Dynamic Programming algorithm for the Optimal Binary Search Tree Problem.

Course Code:

Course Name: Big Data Analytics

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

This course gives an overview of Big Data, i.e. storage, retrieval and processing of big data. In addition, it also focuses on the "technologies", i.e., the tools/algorithms that are available for storage, processing of Big Data. It also helps a student to perform a variety of "analytics" on different data sets and to arrive at positive conclusions.

Course Outcome:

After learning the course the students should be able to:

COURSE OUTCOMES	
CO1	Describe the significance, structure and sources of Big data.
CO2	Explore avenues for analytical scalability using analytical tools and methods.
CO3	Analyze and Design data with Hadoop tools.
CO4	Analyze and Administrate different operations on Hadoop.
CO5	Apply social networking using map reduction technique.

Course Content:

UNIT-I-Introduction to Big Data:

What is big data? Is the "big" part or the "data" art more important? How is big data different? How is big data more of the same? Risks of big data -why you need to tame big data -the structure of big data- exploring big data, most big data doesn't matter- filtering big data effectively mixing big data with traditional data- the need for standards- today's big data is not tomorrow's big data. What web data reveals, Web data in action? A cross-section of big data sources and the value they hold.

UNIT-II-Data Analysis:

Evolution of analytic scalability – The convergence of the analytic and data environments, massively parallel processing systems, Cloudcomputing, Grid computing, Map reduce, Enterprise analytic sand box, Enterprise analytic data sets. Analytic Tools and Methods – The evolution of analytic tools and methods. Analysis approaches – Framing the problem, Statistical significance versus business importance. Enabling Analytic innovation – traditional approaches hamper innovation,

UNIT-III-Map Reduce and Hadoop Distributed File system:

A Weather Dataset, Analyzing the Data with Unix Tools, Analyzing the Data with Hadoop, Scaling Out, Hadoop Streaming, Hadoop Pipes, The Design of HDFS, HDFS Concepts, The Java Interface, Data Flow, Parallel Copying with distcp, Hadoop Archives.

UNIT-IV-Introduction to Hadoop and its Operations:
Administering hadoop- HDFS, Monitoring, and Maintenance. Pig- Installing and running pig, Comparison with Databases, pig latin, User-defined functions. Hive- Installing Hive, running hive, Comparison with traditional databases, HiveQL, Querying data.

UNIT-V-Recommendation Systems and Mining Social- Network Graphs: A model for recommendation systems, Content- based recommendation, Collaborative filtering, Dimensionality Reduction, The Netflix challenge. Mining social-network graphs- Social networks as graphs, Clustering of social-network graphs, Partitioning of graphs, Neighborhood properties of graphs.

Reference Books:

1 Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced
Analytics", John Wiley & sons, 2014.
2 Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets" 2nd edition, CambridgeUniversity Press, 2016.
3 Tom White, "Hadoop: The Definitive Guide", O'reily Media, 4th edition, 2015.
4 Paul Zikopoulos, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill Professional, 2012.
5 Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, Pete Warden, Big Data Glossary, O'Reilly,2016.
6 Chuck Lam, "Hadoop in Action", Dream tech Press, 2nd edition 2014.

Course Code:

Course Name: Foundation of Cyber Security

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

This course involves computational methods providing secure Internet communication. Among the topics covered are: Security threats in communication systems; information economics and incident response etc.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Reasonable understanding of the fundamentals of the cybersecurity domain and related issues.
CO2	Practical knowledge of various tools, processes and methods to ensure security of systems through a minimum of two hands-on assignments involving attack and protection in a virtual environment
CO3	An understanding of the inter-disciplinary nature of cyber security domain.
CO4	Adequate level of cross-disciplinary knowledge of design, implementation, evaluation and testing of secure protocols, systems or applications.

Course Content:

UNIT-I- Introduction to Cybercrime:

Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals? Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens. Cyber offenses: How Criminals Plan Them: How Criminals Plan the Attacks, Social Cyber stalking Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector.

UNIT-II- Cybercrime: Mobile and Wireless Devices:

Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era.

UNIT-III- Tools and Methods Used in Cybercrime:

Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks,

SQL Injection, Buffer Overflow, Attacks on Wireless Networks. Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).

UNIT-IV- Understanding Computer Forensics:

Introduction, Historical Background of Cyber forensics, Digital Forensics Science, The Need for Computer Cyber forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Setting up a Computer Forensics Laboratory: Understanding the Requirements, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics, Forensics and Social Networking Sites: The Security/Privacy Threats, Computer Forensics from Compliance Perspective, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing, Anti forensics.

UNIT-V- Cybercrime and Cyberterrorism:

Social, Political, Ethical and Psychological Dimensions, Introduction, Intellectual Property in the Cyberspace, The Ethical Dimension of Cybercrimes, The Psychology, Mind set and Skills of Hackers and Other Cybercriminals, Sociology of Cybercriminals, Information Warfare: Perception or an Eminent Reality? Cybercrime: Illustrations, Examples and Mini-Cases: Introduction, Real-Life Examples, Mini-Cases, Illustrations of Financial Frauds in Cyber Domain, Digital Signature-Related Crime Scenarios, Digital Forensics Case Illustrations, Online Scams.

Reference Books:

1 Fundamentals of Cyber Security (Principles Theory & Practices), Bhushan Mayank, Publisher: BPB Publications, 2020.
2 Cyber Security Fundamentals, Rajesh Kumar Gautam, BPB Publications, 2019.
3 Cybersecurity: Managing Systems, Conducting Testing, and Investigating Intrusions, Thomas J. Mowbray, Copyright © 2014 by John Wiley & Sons, Inc, ISBN: 91-118 - 84965 -1.
4 Dreamtech Press. ISBN: 9789351194736, 2015 Martti Lehto, Pekka Neittaanmäki, "Cyber Security: Analytics, Technology and Automation edited" Springer International Publishing Switzerland, 2015.
5 Cyber Security Essentials, James Graham, Ryan Olson, Rick Howard, CRC Press, 15 Dec 2010.

Course Code:
Course Name: Introduction to Cloud Computing

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

Apply Cloud Computing and Related Technologies, Applying virtualization concept in reality

Course Outcome:

After learning the course the students should be able to:

COURSE OUTCOMES	
CO1	Understanding of Cloud Computing and Related Technologies
CO2	Appreciate the requirements of various service paradigms in Cloud Computing

Course Content:

UNIT-I- Introduction to Cloud Computing:

Introduction to Cloud Computing, Definition, Characteristics, Components, Cloud provider, SLA, Virtualization, Types of virtualization, Server virtualization, storage virtualization, Network Virtualization and application virtualization, Importance of virtualization in cloud,Study of hypervisors.

UNIT-II- Cloud deployment models:

Cloud deployment models: Public cloud, Private cloud and Hybrid cloud- Organizational scenarios of clouds, , Deploy application over cloud-Workload distribution, Resource pooling, dynamic scalability, elasticity, Service load balancing, Cloud bursting, Service Technology: SOAPand REST Web services, AJAX and mashups Web services, Service Middleware.

UNIT-III- Overview:

Overview, Infrastructure as a Service (IaaS) Cloud Delivery Model, Platform as a Service (PaaS) Cloud Delivery Model, Software as a Service (SaaS) Cloud Delivery Model- Administering& Monitoring cloud services, benefits and limitations- Cloud computing platforms: Infrastructure asa service: Amazon EC2, Platform as a Service: Google App Engine, Microsoft Azure, Utility Computing, Elastic Computing.

UNIT-IV- GFS and HDFS:

GFS and HDFS, Big Table, HBase and Dynamo, Map-Reduce: The Map-Reduce model-Cloud Workload Overview, Workloads most suitable for Cloud, Workloads not suitable for Cloud.

UNIT-V- Tools and Technologies for Cloud:

Tools and Technologies for Cloud, Cloud Computing Platform: Eucalyptus, Nimbus, Open Nebula,Cloud

Mashups, Cloud Tools: VMWare, Eucalyptus, Cloud Sim, Implementing real time application over cloud platform, QOS Issues in Cloud, data migration, streaming in Cloud, Concepts in Mobile Cloud Computing, Fog Computing, Dockers, Green Cloud, Cloud Computing,IoT Cloud.

Reference Books:

1 Thomas Erl, Zaigham Mahmood, and Ricardo Puttini,"Cloud Computing Concepts, Technology & Architecture", Prentice Hall, 2013.

2 A.Srinivasan,J.Suresh,"Cloud Computing, A practical approach for learning and implementation",Pearson,2014.

3 Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, JamesBroberg, Andrzej M. Goscinski, Wiley,2011.

Course Code:

Course Name: Artificial Intelligence and Robotics

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30		-	70	100

Course Description:

This course aims to enable learners to understand about the functional elements of robotics. They will also develop knowledge and skill in inverse and direct kinematics.

Course Outcome:

Upon successful completion of this course students will be able to:

COURSE OUTCOMES	
CO1	Find appropriate idealizations for converting real world problems into AI problems formulated using the appropriate search algorithm.
CO2	Formulate and implement the appropriate search algorithms to find the solutions for real time and heuristics problems.
CO3	Represent and debug knowledge in an appropriate first order logic representation with the understanding of the fundamentals of knowledge representation.
CO4	Choose and Implement the appropriate algorithms for a real world supervised learning Problem .
CO5	Inculcate the basic knowledge of Robotics along with the Artificial Intelligence

Course Content:

UNIT-I- Introduction to AI, Informed Search and Exploration:

Intelligent Agents: Agents and environment; Rationality; the nature of environment; the structure of agents. Problem-solving: Problem solving agents; Example problems; Searching for solution; uninformed search strategies. Informed search strategies; Heuristic functions; On-line search agents and unknown environment.

UNIT-II- Constraint Satisfaction, Adversial Search, Logical Agents:

Constraint satisfaction problems; Backtracking search for CSPs. Adversial search: Games; Optimal decision on games; Alpha-Beta pruning. Knowledge-based agents; The wumpus world as an example world; Logic; proposition logic Reasoning pattern sin proposition logic.

UNIT-III- First-Order Logic, Inference in First-Order Logic:
Representation revisited Syntax and semantics of first-order logic Using first-order logic; Knowledge engineering in first-order logic. Propositional versus first-order inference; Unification and lifting Forward chaining Backward chaining resolution.

UNIT-IV- Knowledge Representation and Learning, AI: Present and Future:
Onto logical engineering; Categories and objects; Actions, situations, and events; Mental events and mental objects; The Internet shopping world; Reasoning systems for categories; Reasoning with default information; Truth maintenance systems. Learning: Forms of Learning; Inductive learning; Learning decision trees; Ensemble learning; Computation all learning theory. AI: Present and Future: Agent components.

UNIT-V- Introduction to Robotics:
Introduction; Robot Hardware: sensors and Effectors; Robotic Perception: localization, mapping, other types of perception; Planning to Move: configuration space, cell decomposition methods and skeletonization methods; Planning uncertain movements: robust methods; Moving: dynamics and control, potential field control and reactive control; Application domains.

Reference Books:

1. Stuart Russel, Petr Norvig: Artificial Intelligence A Modern Approach, 2nd Edition,Pearson Education, 2003.
2. Elaine Rich, Kevin Knight: Artificial Intelligence, 2nd Edition, Tata McGraw Hill,1991.
3. Nils J. Nilsson: Principles of Artificial Intelligence, Elsevier, 1980

Course Code:

Course Name: Web Technologies

Credit Scheme			Evaluation Scheme				
Lecture	Practical	Credit	Internal			External	Total
			Written	Practical	Tutorial		
5 Hrs./Week	-	4	30	-	-	70	100

Course Description:

This course gives an overview of below topics:

1. To learn HTML tags and JavaScript Language programming concepts and techniques.
2. To develop the ability to logically plan and develop web pages.
3. To learn to write, test, and debug web pages using HTML and JavaScript.
4. To Implement basic of MEAN and FULL stack.

Course Outcome:

After learning the course the students should be able to:

COURSE OUTCOMES	
CO1	Apply the concept and usages of web-based programming techniques.
CO2	Demonstrate the development of XHTML documents using JavaScript and CSS.
CO3	Design and implement user interactive dynamic web-based applications.
CO4	Demonstrate applications of AngularJS and jQuery for the given problem.
CO5	Implement modern web applications using MEAN & FULL stack..

Course Content:

UNIT-I- OVERVIEW OF WEB TECHNOLOGIES:

HTML5 and CSS: Web browsers, web servers, MIME, HTTP Introduction to XHTML5 tags, Basic syntax and structure, text markups, images, lists, tables, Media tags-audio and video, forms, frames. CSS: Introduction to CSS, Levels of CSS, Selectors, Font, color and Text Properties, BOX Model, Span and Div tags. Introduction to JavaScript, controls statements, Arrays and functions, pattern matching, Element Access, Event Handling.

UNIT-II- XML and AJAX:

XML-Documents and Vocabularies-Versions and Declaration - Namespaces JavaScript and XML: Ajax-DOM based XML processing Event-oriented Parsing: SAX-Transforming XML Documents Selecting XML Data: XPATH-Template basedTransformations: XSLT-Displaying XML Documents in Browsers - Evolution of AJAX -Webapplications with AJAX -AJAX Framework.

UNIT-III- CLIENT-SIDE SCRIPTING:

Introduction to JQuery, Syntax, selectors, events, JQuery HTML, JQuery Effects, JQuery CSS. Introduction to Angular JS, Directives, Expressions, Directives, Controllers, Filters, Services, Events, Forms, Validations, Examples. Bootstrap: Introduction to Bootstrap, First example, containers, Bootstrap elements: colors, tables, images, buttons, button groups, progress bars, Forms, utilities, Classes, alerts, custom forms, Grid System.

UNIT-IV- SERVER-SIDE SCRIPTING:

Essentials of PHP Installation of Web Server, XAMPP Configurations-PHP Forms- GET and POST method - Regular Expressions-Cookies- Sessions- Usage of Include and require statements- File: read and write from the file- PHP Filters-PHP XML Parser-Introduction to Node.js-Node.js Modules and filesystem- Node.js Events.

UNIT-V- MySQL and MEAN STACK:

PHP with MySQL- Performing basic database operation (DML) (Insert, Delete, Update, Select)- Prepared Statement- Uploading Image or File to MySQL- Retrieve Image or File from MySQL- Uploading Multiple Files to MySQL- Introduction to MEAN and FULL Stack-Real time example for modern web applicationsusing MEAN-MEAN vs Full Stack.

Reference Books:

1. Christopher Murphy, Richardclark, OliStudholme, DivysManian, "Beginning HTML5 and CSS3", Apress Publication,2012.
2. Grant, Andrew, "Beginning AngularJS", Apress Publication,2014.
3. Matt Doyle, "Beginning PHP 5.3", Wiley Publisher,2010.
4. Chris Bates, "Web Programming Building Internet Applications", Wiley India, 3rd Edition, 2007.
5. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, Pete Warden, Big Data Glossary, O"Reilly,2016.