Annamalai University

Faculty of Science

Department of Computer and Information Science

MASTER OF COMPUTER APPLICATIONS

Programme Code: SCIS31

Regulations, Curricula and Syllabi

(2019-20)

Annamalai University

Faculty of Science

DEPARTMENT OF COMPUTER AND INFORMATION SCIENCE

MASTER OF COMPUTER APPLICATIONS

Programme Code:SCIS31

These rules and regulations shall govern the Three year post graduate studies leading to the award of degree of **Master of Computer Applications** in the Faculty of Science. These academic Regulations shall be called "**Annamalai University, Faculty of Science Three year Master of Computer Applications Regulations 2019**". They shall come into force with effect from the academic year 2019 – 2020.

1. Definitions and Nomenclature

1.1 **University** refers to Annamalai University.

1.2 **Department** means any of the academic departments and academic centers at the University.

1.3 **Discipline** refers to the specialization or branch of knowledge taught and researched in higher education. For example, Computer Science is a discipline in the Mathematical Sciences, while Economics is a discipline in Social Sciences.

1.4 **Programme** encompasses the combination of courses and/or requirements leading to a Degree. For example, MCA, M.Sc., M.A.

1.5 **Course** is an individual subject in a programme. Each course may consist of Lectures/ Laboratory /Seminar/Project work/viva-voce etc. Each course has a course title and is identified by a course code.

1.6 **Curriculum** encompasses the totality of student experiences that occur during the educational process.

1.7 **Syllabus** is an academic document that contains the complete information about an academic programme and defines responsibilities and outcomes. This includes course information, course objectives, policies, evaluation, grading, learning resources and course calendar.

1.8 **Academic Year** refers to the annual period of sessions of the University that comprises two consecutive semesters.

1.9 **Semester** is a half-year term that lasts for a minimum duration of 90 days. Each academic year is divided into two semesters.

1.10 **Choice Based Credit System**: A mode of learning in higher education that enables a student to have the freedom to select his/her own choice of elective courses across various disciplines for completing the Degree programme.

1.11 **Core Course** is mandatory and an essential requirement to qualify for the Degree.

1.12 **Elective Course** is a course that a student can choose from a range of alternatives.

1.13 **Value-added Courses** are optional courses that complement the students' knowledge and skills and enhance their employability.

1.14 **Credit** refers to the quantum of course work in terms of number of class hours in a semester required for a programme. The credit value reflects the content and duration of a particular course in the curriculum.

1.15 Credit Hour refers to the number of class hours per week required for a course in a semester. It is used to calculate the credit value of a particular course.

1.16 Programme Outcomes (POs) are statements that describe crucial and essential knowledge, skills and attitudes that students are expected to achieve and can reliably manifest at the end of a programme.

1.17 Programme Specific Outcomes (PSOs) are statements that list what the graduate of a specific programme should be able to do at the end of the programme.

1.18 Learning Objectives are statements that define the expected goal of a course in **Course Objectives** in terms of demonstrable skills or knowledge that will be acquired by a student.

1.19 Course Outcomes (COs) are statements that describe what students should be able to achieve/demonstrate at the end of a course. They allow follow-up and measurement of learning objectives.

1.20 Grade Point Average (GPA) is the average of the grades acquired in various courses that a student has taken in a semester. The formula for computing GPA is given in section 11.3

1.21 Cumulative Grade Point Average (CGPA) is a measure of overall cumulative performance of a student over all the semesters. Calculation of CGPA is given in section 11.4.

1.22 Letter Grade is an index of the performance of a student in a particular course. Grades are denoted by the letters S, A, B, C, D, E, RA, and W.

2. Programme Offered and Eligibility Criteria:

The Department of Computer and Information Science offers a Three Year Master of Computer Applications programme. The eligibility criteria and admission procedure are followed as per the guidelines of Tamil Nadu Common Entrance Test (TANCET).

3. Reservation Policy: Admission to the various programmes will be strictly based on the reservation policy of the Government of Tamil Nadu.

4. Programme Duration

4.1 The Three Year Master's Programme consists of three academic years.

4.2 Each academic year is divided into two semesters, the first being from July to November and the second from December to April.

4.3 Each semester will have 90 working days (18 weeks).

5. Programme Structure

5.1 The Three Year Master's Programme consists of Core Courses, Elective Courses (Departmental & Interdepartmental), and Project.

5.2 Core courses

5.2.1 These are a set of compulsory courses essential for each programme.

5.2.2 The core courses include both Theory (Core Theory) and Practical (Core Practical) courses.

5.3 Elective courses

5.3.1 Departmental Electives (DEs) are the Electives that students can choose from a range of Electives offered within the Department.

5.3.2 Interdepartmental Electives (IDEs) are Electives that students can choose from amongst the courses offered by other departments of the same faculty as well as by the departments of other faculties.

5.4 Experiential Learning

5.4.1 Experiential learning provides opportunities to students to connect principles of the discipline with real-life situations.

5.4.2 In-plant training/field trips/internships/industrial visits fall under this category.

5.4.3 Experiential learning is categorized as Core.

5.5 Project

5.5.1 Each student shall undertake a Project and submit a dissertation as per guidelines in the final semester.

5.5.2 The Head of the Department shall assign a Research Supervisor to the student.

5.5.3 The Research Supervisor shall assign a topic for research and monitor the progress of the student periodically.

5.5.4 Students who wish to undertake project work in recognized institutions/industry shall obtain prior permission from the University. The Research Supervisor will be from the host institute.

5.6 Value added Courses (VACs)

5.6.1 Students may also opt to take Value added Courses beyond the minimum credits required for award of the Degree. VACs are outside the normal credit paradigm.

5.6.2 These courses impart employable and life skills. VACs are listed in the University website and in the Handbook on Interdepartmental Electives and VACs.

5.6.3 Each VAC carries 2 credits with 30 hours of instruction. Classes for a VAC are conducted beyond the regular class hours and preferably in the III Semester.

5.7 Online Courses

5.7.1 The Heads of Departments shall facilitate enrolment of students in Massive Open Online Courses (MOOCs) platform such as SWAYAM to provide academic flexibility and enhance the academic career of students.

5.7.2 Students who successfully complete a course in the MOOCs platform shall be exempted from one elective course of the programme.

5.8 Credit Distribution: The credit distribution is organized as follows:

	Credits
Core Courses	88
Allied Courses	08
Elective Courses	18
Project	22
Total	**136**

5.9 Credit Assignment

Each course is assigned credits and credit hours on the following basis:

1 Credit is defined as

1 Lecture period of one hour duration per week over a semester

1 Tutorial period of one hour duration per week over a semester

1 Practical / Project period of two hours duration per week over a semester.

6 Attendance

6.1 Each faculty handling a course shall be responsible for the maintenance of Attendance and Assessment Record for candidates who have registered for the course.

6.2 The Record shall contain details of the students' attendance, marks obtained in the

Continuous Internal Assessment (CIA) Tests, Assignments and Seminars. In addition the Record shall also contain the organization of lesson plan of the Course teacher.

6.3 The record shall be submitted to the Head of the Department and Dean once a month for monitoring the attendance and syllabus coverage.

6.4 At the end of the semester, the record shall be placed in safe custody for any future verification.

6.5 The Course teacher shall intimate to the Head of the Department at least seven calendar days before the last instruction day in the semester about the attendance particulars of all students.

6.6 Each student shall have a minimum of 75% attendance in all the courses of the particular semester failing which he or she will not be permitted to write the End-Semester Examination. The student has to redo the semester in the next year.

6.7 Relaxation of attendance requirement up to 10% may be granted for valid reasons such as illness, representing the University in extracurricular activities and participation in NCC/NSS/YRC/RRC.

7   Mentor-Mentee System

7.1 To help the students in planning their course of study and for general advice on the academic programme, the Head of the Department will attach certain number of students to a member of the faculty who shall function as a Mentor throughout their period of study.

7.2 The Mentors will guide their mentees with the curriculum, monitor their progress, and provide intellectual and emotional support.

7.3 The Mentors shall also help their mentees to choose appropriate electives and value-added courses, apply for scholarships, undertake projects, prepare for competitive examinations such as NET/SET, GATE etc., attend campus interviews and participate in extracurricular activities.

8   Examinations

8.1 The examination system of the University is designed to systematically test the student's progress in class, laboratory and field work through Continuous Internal Assessment (CIA) Tests and End-Semester Examination (ESE).

8.2 There will be two CIA Tests and one ESE in each semester.

8.3 The Question Papers will be framed to test different levels of learning based on Bloom's taxonomy viz. Knowledge, Comprehension, Application, Analysis, Synthesis and Evaluation/Creativity.

8.4   Continuous Internal Assessment Tests

8.4.1 The CIA Tests shall be a combination of a variety of tools such as class tests, assignments and seminars. This requires an element of openness.

8.4.2 The students are to be informed in advance about the assessment procedures.

8.4.3 The setting of question paper will be decided by the respective faculty.

8.4.4 CIA Tests will be for one or two hours duration depending on the quantum of syllabus.

8.4.5 A student cannot repeat the CIA Test-I and CIA Test-II. However, if for any valid reason, the student is unable to attend the test, the prerogative of arranging a special test lies with the teacher in consultation with the Head of the Department.

8.4.6 For the CIA Tests, the assessment will be done by the Course teacher

8.5   End Semester Examinations (ESE)

8.5.1 The ESE for the first and third semester will be conducted in November and for the

second and fourth semester in May.

8.6 Candidates who failed in any course will be permitted to reappear in failed course in the subsequent examinations.

8.7 The ESE will be of three hours duration and will cover the entire syllabus of the course.

9 Evaluation

9.1 Marks Distribution

9.1.1 For each course, the Theory and Practical shall be evaluated for a maximum of 100 marks.

9.1.2 For the theory courses, CIA Tests will carry 25% and the ESE 75% of the marks.

9.1.3 For the Practical courses, the CIA Tests will carry 40% and the ESE 60% of the marks.

9.2 Assessment of CIA Tests

9.2.1 For the CIA Tests, the assessment will be done by the course instructor

9.2.2 For the Theory Courses, the break-up of marks shall be as follows

CIA for Theory	Marks
Test-I & Test-II	15
Seminar	05
Assignment	05
Total	25

9.2.3 For the Practical Courses (wherever applicable), the break-up of marks shall be as follows:

CIA for Practical	Marks
Test-I	15
Test-II	15
Viva-voce and Record	10
Total	40

9.3 Assessment of End-Semester Examinations

9.3.1 Evaluation for the ESE is done by internal examiner.

9.4 Assessment of Project/Dissertation

9.4.1 The Project Report/Dissertation shall be submitted as per the guidelines.

9.4.2 The Project Work/Dissertation shall carry a maximum of 200 marks.

9.4.3 CIA for Project work/dissertation will carry 40% and ESE 60%.

9.4.4 The Project Report evaluation and viva-voce will be conducted by a committee constituted by the Head of the Department.

9.4.5 The Project Evaluation Committee will comprise the Head of the Department, Project Supervisor, and a senior faculty.

9.4.6 **The marks shall be distributed as follows:**

9.5 **Assessment of Value-added Courses**
9.5.1 Assessment of VACs shall be internal. Two CIA Tests shall be conducted during the semester by the Department(s) offering VAC.
9.5.2 The grades obtained in VACs will not be included for calculating the GPA/CGPA.
9.6 **Passing Minimum**
9.6.1 A student is declared to have passed in each course if he/she secures not less than 40% marks in the ESE and not less than 50% marks in aggregate taking CIA and ESE marks together.
9.6.2 A candidate who has not secured a minimum of 50% of marks in a course (CIA + ESE) shall reappear for the course in the next semester/year.
10. **Conferment of the Master's Degree**

A candidate who has secured a minimum of 50% marks in all courses prescribed in the programme and earned the minimum required credits shall be considered to have passed the Master's Programme.

11. **Marks and Grading**
11.1 The performance of students in each course is evaluated in terms Grade Point (GP).

Continuous Internal Assessment (50 Marks)		End Semester Examination (150 Marks)	
Review-I – 20	Review-II -30	Project / Dissertation Evaluation	Viva-voce
		100	50

11.2 The sum total performance in each semester is rated by Grade Point Average (GPA) while Cumulative Grade Point Average (CGPA) indicates the Average Grade Point obtained for all the courses completed.

11.3 **The GPA** is calculated by the formula

$$GPA = \frac{\sum_{i=1}^{n} C_i G_i}{\sum_{i=1}^{n} C_i}$$

where, C_i is the Credit earned for the Course i in any semester;

G_i is the Grade Point obtained by the student for the Course i and

n is the number of Courses passed in that semester.

11.4 **CGPA** is the Weighted Average Grade Point of all the Courses passed starting from the first semester to the current semester.

$$CGPA = \frac{\sum_{i=1}^{m}\sum_{i=1}^{n} C_i G_i}{\sum_{i=1}^{m} \sum_{i=1}^{n} C_i}$$

Where, C_i is the Credit earned for the Course i in any semester;

G_i is the Grade Point obtained by the student for the Course i

n is the number of Courses passed in that semester.

m is the number of semesters.

11.5 Evaluation :

11.5.1 Performance of the student for each course will be rated as shown in the Table.

Range of Marks	Grade Points	Letter Grade
90 and above	10	S
80-89	9	A
70-79	8	B
60-69	7	C
55-59	6	D
50-54	5	E
Less than 50	0	RA
Withdrawn from the examination	0	W

11.5.2 A ten point rating scale is used for evaluation of the performance of the student to provide overall grade for the Master's Programme.

CGPA	Classification of Final Result
8.25 and above	First Class with Distinction
6.5 and above but below 8.25	First Class
5.0 and above but below 6.5	Second Class
0.0 and above but below 5.0	Re-appear

11.6 Classification of Results. The successful candidates are classified as follows:

11.6.1 **For First Class with Distinction:** Candidates who have passed all the courses prescribed in the Programme in the first attempt with a CGPA of 8.25 and above within the programme duration. Candidates who have withdrawn from the End Semester Examinations are still eligible for First Class with Distinction (See Section 12 for details).

11.6.2 **For First Class:** Candidates who have passed all the courses with a CGPA of 6.5 and above.

11.6.3 **For Second Class:** Candidates who have passed all the courses with a CGPA between 5.0 and less than 6.5.

11.6.4 Candidates who obtain overall highest CGPA in all examinations in the first appearance itself are eligible for University Rank.

11.7 Course-Wise Letter Grades

11.7.1 The percentage of marks obtained by a candidate in a course will be indicated in a letter grade.

11.7.2 A student is considered to have completed a course successfully and earned the credits if he/she secures an overall letter grade other than RA.

11.7.3 A course successfully completed cannot be repeated for the purpose of improving the Grade Point.

11.7.4 A letter grade RA indicates that the candidate shall reappear for that course. The RA

Grade once awarded stays in the grade card of the student and is not deleted even when he/she completes the course successfully later. The grade acquired later by the student will be indicated in the grade sheet of the Odd/Even semester in which the candidate has appeared for clearance of the arrears.

11.7.5 If a student secures RA grade in the Project Work/Field Work/Practical Work/Dissertation, he/she shall improve it and resubmit if it involves only rewriting/ incorporating the clarifications suggested by the evaluators or he/she can re-register and carry out the same in the subsequent semesters for evaluation.

12. Provision for Withdrawal from the End Semester Examination

12.1 The letter grade W indicates that a candidate has withdrawn from the examination.

12.2 A candidate is permitted to withdraw from appearing in the ESE for one course or courses in ANY ONE of the semesters ONLY for exigencies deemed valid by the University authorities.

12.3 Permission for withdrawal from the examination shall be granted only once during the entire duration of the programme.

12.4 Application for withdrawal shall be considered only if the student has registered for the course(s), and fulfilled the requirements for attendance and CIA tests.

12.5 The application for withdrawal shall be made ten days prior to the commencement of the examination and duly approved by the Controller of Examinations. Notwithstanding the mandatory prerequisite of ten days notice, due consideration will be given under extraordinary circumstances.

12.6 Withdrawal will not be granted for arrear examinations of courses in previous semesters and for the final semester examinations.

12.7 Candidates who have been granted permission to withdraw from the examination shall reappear for the course(s) when the course(s) are offered next.

12.8 Withdrawal shall not be taken into account as an appearance for the examination when considering the eligibility of the candidate to qualify for First Class with Distinction.

13. **Academic misconduct:** Any action that results in an unfair academic advantage/interference with the functioning of the academic community constitutes academic misconduct. This includes but is not limited to cheating, plagiarism, altering academic documents, fabrication/falsification of data, submitting the work of another student, interfering with other students' work, removing/defacing library or computer resources, stealing other students' notes/assignments, and electronically interfering with other students'/University's intellectual property. Since many of these acts may be committed unintentionally due to lack of awareness, students shall be sensitized on issues of academic integrity and ethics.

14. **Transitory Regulations:** Wherever there has been a change of syllabi, examinations based on the existing syllabus will be conducted for two consecutive years after implementation of the new syllabus in order to enable the students to clear the arrears. Beyond that, the students will have to take up their examinations in equivalent subjects, as per the new syllabus, on the recommendation of the Head of the Department concerned.

15. Notwithstanding anything contained in the above pages as Rules and Regulations governing the Three Year Master's Programmes at Annamalai University, the Syndicate is vested with the powers to revise them from time to time on the recommendations of the Academic Council.

Master of Computer Applications (Three Year) Programme
Programme Code: SCIS31

CURRICULA AND SCHEME OF EXAMINATIONS
(For students admitted from the academic year 2019-2020)

Course Code	Course Title	L	T	P	C	CIA	ESE	Total
		Hours/Week			**Credit**	**Marks**		
Semester – I								
19MCAC101	Allied-I: Mathematical Foundations for Computer Applications	4	0	0	4	25	75	100
19MCAC102	Core 1: Object-Oriented Programming using C++	4	0	0	4	25	75	100
19MCAC103	Core 2: Advanced Data Structures and algorithms	4	0	0	4	25	75	100
19MCAC104	Core 3: Computer Organization and Architecture	4	0	0	4	25	75	100
19MCAP106	Core 4: Programming Lab - I (Object-Oriented Programming) using C++)	0	0	4	2	40	60	100
19MCAP107	Core 5: Programming Lab - II (Data Structures using C++)	0	0	4	2	40	60	100
19XXXXXXX	Elective-I: Interdepartmental Elective	3	0	0	3	25	75	100
					23			
Semester – II								
19MCAC201	Allied-II: Accounting and Financial	4	0	0	4	25	75	100
19MCAC202	Core 6: JAVA Programming	4	0	0	4	25	75	100
19MCAC203	Core 7: Relational Database Management System	4	0	0	4	25	75	100
19MCAC204	Core 8: Operating System	4	0	0	4	25	75	100
19MCAC205	Core 9: Software Engineering	4	0	0	4	25	75	100
19MCAP206	Core 10: Programming Lab- III (Java Programming)	0	0	4	2	40	60	100
19MCAP207	Core 11: Programming Lab- IV (RDBMS)	0	0	4	2	40	60	100
					24			
Semester – III								
19MCAC301	Core 12: C# and .Net Frame Work	4	0	0	4	25	75	100
19MCAC302	Core 13: Computer Networks	4	0	0	4	25	75	100
19MCAC303	Core 14: Computer Graphics and Multimedia	4	0	0	4	25	75	100
19MCAC304	Core 15: Software Testing and quality Assurance	4	0	0	4	25	75	100
19MCAP305	Core 16: Programming Lab- V (Software Testing)	0	0	4	2	40	60	100
19MCAP306	Core 17: Programming Lab – V I (C# and .Net Programming)	0	0	4	2	40	60	100
19XXXXXXX	Elective-II: Interdepartmental Elective	3	0	0	3	25	75	100
19PSCI300	Constitution of India	2	0	0	2*	25	75	100
					23			

Semester – IV								
19MCAC401	Core 18: Compiler Design	4	0	0	4	25	75	100
19MCAC402	Core 19: Web Technology	4	0	0	4	25	75	100
19MCAC403	Core 20: Mobile Computing	4	0	0	4	25	75	100
19MCAP404	Core 21: Programming Lab –VII	0	0	4	2	40	60	100
19MCAP405	Core 22: Programming Lab -VIII	0	0	4	2	40	60	100
19MCAE40X	Elective-III: Department Elective	3	0	0	3	25	75	100
19MCAE40X	Elective-IV: Department Elective	3	0	0	3	25	75	100
					22			
Semester – V								
19MCAC501	Core 23: Big Data Analytics	4	0	0	4	25	75	100
19MCAC502	Core 24: Python and R Programming	4	0	0	4	25	75	100
19MCAC503	Core 25: Cloud Computing	4	0	0	4	25	75	100
19MCAP504	Core 26: Programming Lab–IX (Data Analytics: Python and R)	0	0	4	2	40	60	100
19MCAP505	Core 27: Programming Lab–X (Mini Project)	0	0	4	2	40	60	100
19MCAE50X	Elective-V: Department Elective	3	0	0	3	25	75	100
19MCAE50X	Elective-VI: Department Elective	3	0	0	3	25	75	100
					22			
Semester – VI								
19MCAD601	Dissertation and Viva Voce/In plant training	0	0	#	**22**	50	150	200
	Total Credits				**136**			
Value Added Courses								
Online Courses (SWAYAM or MOOC)								

L- Lectures; P- Practical; C- Credits; CIA- Continuous Internal Assessment; ESE- End-Semester Examination

*19PSCI300 = NON CREDIT COMPULSORY COURSE

Student is required to undergo **44 hours per week** of practical work in software development at any Software Industry/Research Establishment of his/her choice.

Note:

1. Students shall take both Department Electives (DEs) and Interdepartmental Electives (IDEs) from a range of choices available. The details of interdepartmental electives are given in the "**Handbook of Interdepartmental Electives-Two Year Programme**" and listed in the University website.

2. Students may opt for any Value-added Courses listed in the University website. The details of Value Added Courses are given in the "**Handbook of Value Added Courses**" and listed in the University website.

ELECTIVE COURSES
DEPARTMENT ELECTIVE COURSES

S. No.	Course Code	Course Title	Hours/ week			C	Marks		
			L	T	P		CIA	ESE	Total
ELECTIVE - III									
1.	19MCAE406	Object-Oriented Analysis and Design(OOAD)	3	0	0	3	25	75	100
2.	19MCAE407	Open Source Technologies	3	0	0	3	25	75	100
ELECTIVE - IV									
3.	19MCAE408	Soft Computing	3	0	0	3	25	75	100
4.	19MCAE409	Data Mining Techniques	3	0	0	3	25	75	100
ELECTIVE - V									
5.	19MCAE506	Network Security	3	0	0	3	25	75	100
6.	19MCAE507	Internet of Things	3	0	0	3	25	75	100
ELECTIVE - VI									
7.	19MCAE508	Digital Image Processing	3	0	0	3	25	75	100
8.	19MCAE509	Pattern Recognition	3	0	0	3	25	75	100

VALUE ADDED COURSES (VAC)(For students of other departments)

To be offered in THIRD Semester

S. No.	Course Code	Course Title	Hours/ week			C	Marks		
			L	T	P		CIA	ESE	Total
1.	SCISVAC01	Web Development	3	0	0	3	25	75	100
2.	SCISVAC02	Fundamentals of Computing	3	0	0	3	25	75	100
3.	SCISVAC03	Advanced Web Development	3	0	0	3	25	75	100
4.	SCISVAC04	Internet and its Applications	3	0	0	3	25	75	100

Inter- Department Elective Course (IDE) (Offered by other departments)

S.No	Course Code	Course Title	Department	L	P	Credits	CIA	ESE	Total
				Hrs				Marks	
1.	19AENGX01	Soft Skills	**English**	3	0	3	25	75	100
2.	19SMATX01	Discrete Mathematics	**Mathematics**	3	0	3	25	75	100
3.	19SMATX02	Numerical Methods		3	0	3	25	75	100
4.	19SMATX03	Statistical Computing		3	0	3	25	75	100
5.	19SMATX04	Resource Management Techniques		3	0	3	25	75	100
6.	19SMATX05	Differential Equations		3	0	3	25	75	100
7.	19SSTAX01	Statistical Methods	**Statistics**	3	0	3	25	75	100
8.	19SSTAX02	Mathematical Statistics		3	0	3	25	75	100
9.	19SSTAX03	Bio-Statistics		3	0	3	25	75	100
10.	19SPHYX01	Classical Mechanics and Special Theory of Relativity	**Physics**	3	0	3	25	75	100
11.	19SPHYX02	Physics of the Earth		3	0	3	25	75	100
12.	19SPHYX03	Bio-Medical Instrumentation		3	0	3	25	75	100
13.	19SCHEX01	Applied Chemistry	**Chemistry**	3	0	3	25	75	100
14.	19SCHEX02	Basic Chemistry		3	0	3	25	75	100
15.	19SCHEX03	Instrumental Methods of Chemical Analysis		3	0	3	25	75	100
16.	19SBOTX01	Plant Tissue Culture	**Botany**	3	0	3	25	75	100

17.	19SBOTX02	Plant Science – I		3	0	3	25	75	100
18.	19SBOTX03	Gardening and Horticulture		3	0	3	25	75	100
19.	19SBOTX04	Plant Science – II		3	0	3	25	75	100
20.	19SZOOX01	Animal Culture Techniques	**Zoology**	3	0	3	25	75	100
21.	19SZOOX02	Environmental Science		3	0	3	25	75	100
22.	19SGEOX01	Environmental Geosciences	**Earth Science**	3	0	3	25	75	100
23.	19SGEOX02	Applied Geophysics		3	0	3	25	75	100
24.	19SBIOX01	Basic Biochemistry	**Biochemistry & Biotechnology**	3	0	3	25	75	100
25.	19SBIOX02	Basic Biotechnology		3	0	3	25	75	100

ANNAMALAI UNIVERSITY

Department of Computer and Information Science

Pattern of question paper for END semester examinations

(Based on Revised Bloom's Taxonomy)

Year : I and II

Programme: MCA Three Year PG Programme   **Semester: I / II / III / IV**

Course Code: **Course Name:**

Time: 3 Hrs   **Max.Marks:100**

Part-A (Level-K1/ Level-K2) Marks: (10x2=20)
(Answer ALL of the questions)

1. Define……
2. Multiple Choices   a.   b.   c.   d.
3. Multiple Choices   a.   b.   c.   d.
4. Match the following   i - a   ii- b   iii- c   iv –d   v - …..
5. Match the following   i - a   ii- b   iii- c   iv –d   v - …..
6. Explain…….
7. Select…..
8. Describe……
9. Classify….
10. Elucidate….

Part-B (Level-K3/ Level-K4)Marks: (8x5=40)
(Answer any EIGHT of the questions)

11. Prepare…..
12. Solve…..
13. Apply……
14. Show…..
15. Categorize…
16. Analyze…
17. Distinguish….
18. Infer….
19. Compare….
20. Compute

Part-C (Level-K5)Marks: (3x10=30)
(Answer any THREE of the questions)

21. Discuss…
22. Summarize….
23. Evaluate…..
24. Disprove….

Part-D (Level-K6)*Marks: (1x10=10)
(Answer any ONE of the questions)

25. Design….
26. Develop…

ANNAMALAI UNIVERSITY
Department of Computer and Information Science
Year : III

Programme: MCA Three Year PG Programme   Semester: V

Course Code: Course Name:
Time: 3 Hrs   Max.Marks:100

Part-A (Level-K1/ Level-K2)Marks: (10x2=20)
(Answer ALL of the questions)

1. Define……
2. Multiple Choices   a.   b.   c.   d.
3. Multiple Choices   a.   b.   c.   d.
4. Match the following   i - a   ii- b   iii- c   iv –d   v - …..
5. Match the following   i - a   ii- b   iii- c   iv –d   v - …..
6. Explain…….
7. Select…..
8. Describe…..
9. Classify….
10. Elucidate….

Part-B (Level-K3/ Level-K4)Marks: (6x5=30)
(Answer any SIX of the questions)

11. Apply……
12. Show…..
13. Prepare
14. Make use of….
15. Categorize…
16. Analyze…
17. Distinguish….
18. Simplify…..

Part-C (Level-K5)Marks: (3x10=30)
(Answer any THREE of the questions)

19. Discuss…
20. Recommend with
21. Evaluate…..
22. Justify….
23. Optimize…

Part-D (Level-K6)*Marks: (2x10=20)
(Answer any TWO of the questions)

24. Design….
25. Formulate …
26. Modify …..

Master of Computer Applications (THREE YEAR) Programme								
[End Semester Examinations] Bloom's Taxonomy - Questions Conforming to Levels K1 to K6								
I & II Year (Three year PG)					III Year (Three Year PG)			
Level	Part	Questions & Marks	Total Marks		Level	Part	Questions & Marks	Total Marks
K1	A	5 x 2	10		K1	A	5 x 2	10
K2		5 x 2	10		K2		5 x 2	10
K3	B	4 x5	20		K3	B	2 x 5	10
K4		4 x5	20		K4		4 x 5	20
K5	C	3 x 10	30		K5	C	3 x10	30
K6	D	1 x 10	10		K6	D	2x 10	20
			100					100

ANNAMALAI UNIVERSITY
Department of Computer and Information Science
[Question Paper Pattern - INTERNAL TESTS I & II (CIA)]
(Based on Revised Bloom's Taxonomy)

Programme: MCA Three Year PG Programme   **Semester: All**

Time: 2 Hrs   **Max.Marks:50**

Part-A (Level-K1)   Marks: (6x2=12)
(Answer ALL of the questions)

1. Define /Choose/ Relate……
2. What / Why / How?
3. Multiple Choices   a.   b.   c.   d.
4. Multiple Choices   a.   b.   c.   d.
5. Match the following   i - a   ii - b   iii - c   iv - d   v - …..
6. Match the following   i - a   ii - b   iii - c   iv - d   v - …..

Part-B (Level-K2)   Marks: (3x5=15)
(Answer any THREE of the questions)

7. Explain…..
8. Describe…..
9. Select……
10. Compare

Part-C (Level-K3/ Level-K4) Marks: (2x7=14)
(Answer any TWO of the questions)

11. Apply….
12. Calculate….
13. Categorize…

Part-D (Level-K5/ Level-K6)   Marks: (1x9=9)
(Answer any ONE of the questions)

14. Discuss….
15. Summarize….

PROGRAMME OUTCOMES (POs)

After the successful completion of the M.Sc. Computer Science (2 year) Degree Programme, the graduates will be able to:

PO1	An ability to apply knowledge of mathematics, computer science and management in practice.
PO2	An ability to identify, critically analyze, formulate and develop computer applications.
PO3	An ability to select modern computing tools and techniques and use them with dexterity.
PO4	An ability to design a computing system to meet desired needs within realistic constraints such as safety, security and applicability.
PO5	An ability to devise and conduct experiments, interpret data and provide well informed conclusions.
PO6	An ability to understand the impact of system solutions in a contemporary, global, economical, environmental, and societal context for sustainable development
PO7	An ability to function professionally with ethical responsibility as an individual as well as in multidisciplinary teams with positive attitude
PO8	An ability to effectiveness in communicating with a wide range of audiences

PROGRAMME SPECIFIC OUTCOMES (PSOs)

At the end of the programme, the student will be able to

PSO1	To prepare graduates who will productive careers in software industry, corporate sector, Govt. organizations and academia by providing skill based environment for teaching and research in the core and emerging areas of the discipline.
PSO2	To prepare graduates who will contribute to society as broadly educated, expressive, ethical and responsible citizens with proven expertise.
PSO3	To prepare graduates who will achieve peer-recognition; as an individual or in a team; through demonstration of good analytical, design and implementation skills.
PSO4	Develop and deploy software systems with assured quality and efficiency.
PSO5	To prepare graduates who will thrive to pursue life-long learning to fulfill their goals.

19MCAC101: MATHEMATICAL FOUNDATIONS FOR COMPUTER APPLICATIONS

Learning objectives:

- Computer science is the art of solving problems with computers.
- This is a broad definition that encompasses an equally broad field.
- Within computer science, we find software engineering, bioinformatics, cryptography, machine learning, human-computer interaction, graphics, and a host of other fields.
- Mathematics underpins all of these endeavors in computer science.
- To enable the student to learn the mathematical foundations of computer science.

Unit–I

Matrix Algebra: Matrices, Rank of Matrix, Solving System of Equations-Eigen Values and Eigen Vectors- Inverse of a Matrix-Cayley Hamilton Theorem.

Unit-II

Basic Set Theory: Basic Definitions-Venn Diagrams and set operations-Laws of set theory-Principle of inclusion and exclusion-partitions-Permutation and Combination-Relations-Properties of relations - Matrices of relations - Closure operations on relations - Functions-injective, surjective and bijective functions.

Unit-III

Mathematical Logic: Propositions and logical operators - Truth table – Propositions generated by a set, Equivalence and implication –Basic laws-Some more connectives - Functionally completesetofconnectives-Normalforms-ProofsinPropositionalcalculus-Predicate calculus.

Unit–IV

Formal Languages and Finite State Automata: Deterministic finite accepters – Regular languages- Non deterministic finite accepters-Equivalence of deterministic and Non deterministic finite accepters – Reduction of the number of states in finite automata – Regular expressions – Connection between regular expressions and regular languages.

Unit–V

Formal Languages and Finite State Automata (continuation): Closure properties of regular Languages – Identifying non regular Languages-A Pumping Lamma – A context free Languages – context free grammars – leftmost and rightmost derivation – derivation tree – relation between sentential form and derivation tree.

Text Books:

1. KennethH.Rosen,"DiscreteMathematicsandItsApplications",TataMcGrawHill,FourthEdition,2002

.

2.Hopcroft and Ullman, "Introduction to Automata Theory, Languages and Computation",NarosaPublishingHouse,Delhi,2002.

3.Peter Linz, "An Introduction to formal Languages and automata", Jones & Bartlett Publishers,Inc, 2011

References:

1.Tamilarasi A&A.M.Natarajan,"DiscreteMathematicsanditsApplication", KhannaPublishers, 2^{nd}Edition,2005.

2. Venkataraman M K,"EngineeringMathematics", VolumeII,National PublishingCompany, 2^{nd}Edition,1989.

3. John Hopcropt and Jeffrey Ullam, "Introduction to automata Theory, Language, and Computation, Pearson, 3rd Edition, 2006.

Course outcomes:

CO1: Do matrix operation, can find inverse of a given matrix, eigen values and eigen vectors.

CO2: Solve simply problems an probability and baye's theorem.

CO3:Fit straight line by the method of least squares and they will do the problem in correlation and regression analysis for comparing two or more items.

CO4: Solve some basic problems related to automata and logical relations.

CO5: Get numerical solutions to algebraic and transcendental equations and able to solve the system of simultaneous linear equations using different Gaussian methods.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1		✔		✔		✔			✔			✔	
CO2		✔	✔		✔					✔	✔	✔	✔
CO3	✔						✔	✔		✔			✔
CO4	✔	✔		✔	✔				✔	✔	✔	✔	
CO5			✔		✔			✔	✔		✔		

19MCAC102: OBJECT-ORIENTED PROGRAMMING USING C++

Learning Objectives:

- To explain the advantages of object oriented programming over procedure oriented programming.
- Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism.
- To learn how to implement constructors, function, pointes and class member functions.
- Explain array handing, function overloading, operator overloading and virtual functions.
- Helps in implementing some important features of C++ including templates, utilizing the I/O classes in C++ and exception handling.

UNIT-I

Introduction to OOP: Overview of C++ - classes - structures - union - friend function - friend class - inline function - constructors - static members - scope resolution operator - passing objects to functions - function returning objects

UNIT-II

Arrays - pointers - this pointer - references - dynamic memory allocation - functions overloading - default arguments - overloading constructors - pointers to functions

UNIT-III

Operator overloading - member operator function - friend operator function - type conversion - inheritance - types of inheritance - virtual base class - polymorphism - virtual function.

UNIT-IV

Class templates and generic classes - function templates and generic functions -overloading function templates - power of templates - exception handling - derived class exception - exception handling functions

UNIT-V

Streams - formatted I/O with its class functions and manipulators - creating own manipulators - file I/O - conversion functions - standard template library.

Text Book:

1. Balagurusamy E, "Object Oriented Programming with C++", 3/E, TMG, 2006.

References:

1. Hubbard,"Programming with C++", 2/e, Schaum Outline Series, TMH, 2006.

2. Bjarne Stroustrup, "The C++ Programming Language", Addison Wesley Publications,

   Second Edition, 1991.

3. Sarang Proonachandra,"Object Oriented Programming with C++", PHI, 2006.

4. Jagadev A K, Rath A M, and Dehuri S,"Object Oriented Programming Using C++", PHI, 2007.

Course Outcomes:

CO1: Describe the principles of object-oriented problem solving and programming.

CO2: Explain programming fundamentals, including statement and control flow.

CO3: Apply the concepts of class, method, constructor, pointers, data abstraction, function abstraction, inheritance, overriding, overloading, polymorphism, IO streams, Templates.

CO4: Design program with basic data structure like array.

CO5: Develop good quality software using object-oriented techniques.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔				✔			
CO2	✔		✔		✔				✔	✔	✔	✔	✔
CO3	✔						✔	✔		✔			✔
CO4	✔	✔			✔				✔	✔	✔	✔	
CO5				✔		✔	✔		✔		✔		

19MCAC103: ADVANCED DATA STRUCTURES AND ALGORITHMS

Learning Objectives:

- To teach efficient storage mechanisms of data for an easy access.
- To design and implementation of various basic and advanced data structures.
- To introduce various techniques for representation of the data in the real world.
- To develop application using data structures.
- To teach the concept of protection and management of data.

Unit-I

The Role of algorithm in computing - Insertion sort - Analyzing algorithm - Designing algorithm - Divide and Conquer - The maximum-sub array problem - Strasen's algorithm for

matrix multiplication - The substitution method for solving recurrences - The recursion tree method for solving recurrences - Proof of the master theorem.

Unit-II

Sorting and order statistics - Heap sort - Maintaining the heap priority - Building a heap - The heap sort algorithm - Priority queues - Quick sort - Description of quick sort - Performance of quick sort - A randomized version of quick sort - Analysis of quick sort - Sorting in Linear Time - Lower bounds for sorting - Counting sort - Radix sort - Bucket sort - Medians and order statistics - Minimum and Maximum - Selection in expected linear time - Selection in worst case linear time.

Unit-III

Elementary Data Structures - Stacks and Queues - Linked Lists - Implementing pointers and objects - Hash tables - Direct-address tables - Hash functions - Open addressing - Perfect hashing - Binary search trees - Querying binary search trees - Insertion and deletion - Red-Black trees - Properties - Rotations - Insertion - Deletion.

Unit-IV

Advanced Design and Analysis Techniques - Dynamic Programming - Rod cutting - Matrix chain multiplication - Elements of dynamic programming - Longest common subsequence - Optimal binary search trees - Greedy algorithms - An activity-selection algorithm - Elements of greedy strategy - Huffman codes - Matroids and Greedy method - A task scheduling problem as matroid.

Unit-V

Advanced Data Structures - B- trees - Definition - Basic Operations on B-trees - Deleting a key from B-tree - Fibonacci heaps - Structure - Mergeable heap operations - Decreasing a key and deleting a node - Bounding the maximum degree - Van Emde Boas Trees - Preliminaries - Recursive structures - Data structure for disjoint sets - Disjoint set operations - Linked list representation of disjoint sets.

Text Book:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Cliford Stein, Introduction to Algorithms, Third Edition, The MIT press, 2009.

References:

1. Timothy Budd, An Introduction to Object Oriented Programming, Pearson Education, Second Edition, 1996.
2. Jean Paul Tremblay and Paul G. Sorenson, An Introduction to Data Structures with Applications, Tata McGraw Hill, Second Edition, 2010.
3. Sahini, Data Structures, Algorithms and Applications in C++, Tata McGrawHill, 1998.

Course Outcomes:

CO1: Student will be able to choose appropriate data structure as applied to specified problem definition.

CO2: Student will be able to handle operations like searching, insertion, deletion, traversing mechanism etc. on various data structures.

CO3: Students will be able to apply concepts learned in various domains like DBMS, compiler construction etc.

CO4: Students will be able to use linear data structures.

CO5: Students will be able to use non-linear data structures.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔			✔		✔		✔
CO2			✔		✔			✔		✔	✔		✔
CO3	✔			✔			✔						✔
CO4	✔	✔			✔			✔	✔	✔	✔	✔	
CO5		✔	✔			✔	✔				✔	✔	

19MCAC104: COMPUTER ORGANIZATION AND ARCHITECTURE

Learning Objectives:

- To conceptualize the basics of organizational and architectural issues of a digital computer.
- To analyze performance issues in processor of a digital computer.
- To analyze performance issues in memory design of a digital computer.
- To understand various data transfer techniques in digital computer.
- To analyze processor performance improvement using instruction level parallelism Course.

Unit-I

Overview And History- Structure and Functions – Evolution of the Intel x86 architecture – Performance Issues – Computer Components- Bus Interconnection - PCI .

Arithmetic and Logic – Number System – Computer Arithmetic – ALU - Integer Arithmetic – Floating point Arithmetic.

Unit-II

Memory System Design – Characteristics of memory system **–** Memory Hierarchy – Cache Memory Principles – Elements of cache design-Replacement Policies –Main Memory Organization – Optical Memory- Magnetic Tapes.

Input /Output – I/O Module – Programmed I/O – Interrupt Driven I/O – Direct Memory Access (DMA).

Unit-III

Instruction Set Design - Assembly/Machine Language– Addressing Modes- Instruction format –Instruction set design – Type of Operand - Type of Operations –Reduced Instruction Set Computers –RISCVS CISC.

Unit-IV

Processor Structure and Function - Processor Organization – Register Organization – Instruction Cycle – Instruction Pipelining **–**Pipelining Strategy–Pipelining Performance - Pipelining Hazards –Structural Hazards and Data Dependencies–Branch Delay.

Unit-V **Multiprocessors and Multicore computers –** Multiple Processor Organization-SISD -SIMD – MISD and MIMD-Symmetric multiprocessors **–** Cache Coherence – Multicore Organization .

Text Book:

1. Stallings, William., "Computer Organization and Architecture: Designing for Performance", Pearson Education, Tenth Edition, 2016.

References:

1. Hennessy J and D.Patterson, "Computer Architecture- A Quantitative approach "organ and Kafumann, Fifth Edition, 2012.
2. Morris Mano, M, "Computer System Architecture", Pearson, Third Edition, 2016.
3. Govindarajalu B, "Computer Architecture and Organization", Tata Mc GrawHill, Second Edition,2005
4. John.P. Hayes,"Computer Architecture and Organization", Tata McGraw Hill, Third Edition,1998.
5. Hamatcher V.C,etal "Computer Organization", Tata McGraw Hill, 1996.

Course Outcomes:

CO1: Ability to understand basic structure of computer.

CO2: Ability to perform computer arithmetic operations and understand control unit operations.

CO3: Ability to design memory organization that uses banks for different word size operations.

CO4: Ability to understand the concept of cache mapping techniques and I/O organization.

CO5: Ability to conceptualize instruction level parallelism.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1		✔		✔		✔	✔	✔		✔			✔
CO2			✔		✔				✔	✔			✔
CO3	✔	✔		✔			✔			✔			✔
CO4	✔	✔			✔				✔		✔	✔	
CO5			✔	✔		✔		✔	✔			✔	✔

19SOSX105: SOFT SKILLS DEVELOPMENT

Learning objectives:

- To develop the soft skill among the students.
- To impart the student knowledge in developing the positive attitude and art of speaking and writing.
- To indulge the students to improve the body language group discussion and time management.

Unit-I

Soft skills and developing positive Attitude

Soft skills: introduction – what are soft skills? - selling your soft skills - attribute regarded as soft skills – soft skills – social- soft skills- thinking – soft skills –Negotiating –exhibiting your soft skills- indentifying your soft skills- improving your soft skills - soft skills training –train yourself-top 60 soft skills.

Developing positive attitude: introduction – meaning - features of attitudes- attitude and behavior formation of attitudes– change of attitudes – what can you do to change attitude?-ways of changing attitude in a person – attitude in a workplace – the power of positive attitude-developing positive attitude-example of positive attitude- example of negative attitude-over coming negative attitude- negative attitude and its result.

Unit-II

Art of speaking and writing

Art of speaking: Introduction-what make communication important? - Defining communication-special features of communication –communication process- channel of communication- importance of communication - tips for effective communication - tips for powerful presentation-art of public speaking - importance of public speaking.

Art of writing: Introduction – importance of writing –creative writing - writing tips- drawbacks of writing communication.

Unit-III

Body language

Introduction – body talk – voluntary and involuntary body language-forms of body language- parts of body language - origin of body language - uses of body language - body language in building interpersonal relations – body language in building industrial relations-reason to study body language-improving your body language – types of body language-Gender differences- female interest and body language - shaking hands with women - interpreting body language- developing confidence with correct body language.

Unit-IV

Group discussion

Introduction – meaning of GD – why group discussion? - characters tested in a GD – tips on GD – types of GD - skills required in a GD - consequences of GD - behavior of a GD - essential elements of GD - different characters in GD - traits tested in a GD - GD etiquette - areas to be concentrated while preparing for a GD - imitating a GD - techniques to initiate a GD - Non-verbal communication in GD – movement and gestures to be avoided in a GD-topics for GD.

Interview skills

Introduction – why an interview?.- types of interview - interview panel-types of questions asked-reason for selecting a candidate –reason for rejecting a candidate – on the day of interview– on the interview table – attending job fair-common mistakes that you would't want to do-questions the candidate should not ask during the interview –post- interview etiquette- how does one follow up?- telephonic interview –dress code at interview – typical questions asked – interview mistakes –quick tips- how to present well in interview –tips to make a good impression in an interview – job interview-basic tips-how to search for job effectively – interview quotations.

Unit-V

Time management

Introduction- the 80:20 rule- take a good look at the CO1ple around you- examine your work-sense of time management – time is money – features of time- three secretes of time management - time management matrix- analysis of time matrix-effective scheduling – grouping of activities – five steps to successful time management –difficulties in time management- evils of not planning - time management is a myth – overcoming procrastination – ways of find free time- time management tips for students – interesting facts about time-ideal way of spending a day- time wasters – time savers – realizing the value of time-time circle planner.

Text Book:

1. Alex K., "Soft Skills: Know yourself and know the world" S.Chand & company Pvt Lts, Third revised Edition, 2014.

References:

1. Gopalaswamy Ramesh, and Mahadevan Ramesh, " The ACE of Soft Skills, Attitude, Communication and Etiquette for Success", Pearson; First edition, 2013.
2. Barun K. Mitra, "Personality Development and Soft Skills", Oxford university press, New Delhi, 2011.
3. Rao M.S, "Soft Skills - Enhancing Employability: Connecting Campus with Corporate", I K International Publishing House Pvt. Ltd, 2011
4. Sanjay Kumar and Pushp Lata, "Communication Skills", Oxford university press, New Delhi, 2011.

Course Outcomes:

CO1: The students' community enrich the knowledge in the field of soft skills.

CO2: They can able to cope up with recent development in business world.

CO3: The students will have the powerful knowledge in attitude and personality.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✓	✓		✓		✓		✓		✓			
CO2	✓		✓		✓			✓		✓	✓	✓	✓
CO3	✓						✓		✓	✓			✓

19MCAP106 - PROGRAMMING LAB - I

(Object-Oriented Programming using C++)

Learning Objectives:

- At the end of the course students should be familiar with the main features of the C++ language.
- Be able to write a C++ program to solve a well specified problem.
- Understand a C++ program written by someone else.
- Be able to debug and test C++ programs;
- Understand how to read C++ doc library documentation and reuse library code.

- To make the students understand the features of object oriented principles and familiarize them with virtual functions, templates and exception handling.
- To make the students to develop applications using C++.

Lab Exercises

1. Programs on concept of classes and objects.
2. Programs using inheritance.
3. Programs using static polymorphism.
4. Programs on dynamic polymorphism.
5. Programs on operator overloading.
6. Programs on dynamic memory management using new, delete operators.
7. Programs on copy constructor and usage of assignment operator.
8. Programs on exception handling.
9. Programs on generic programming using template function & template class.
10. Programs on file handling.

Course outcomes:

CO1: Students will be able to apply the computer programming techniques to solve practical problems.
CO2: Students will be able to understand the concepts and implementation of constructors and destructors.
CO3: Students will be able to develop software applications using object oriented programming language in C++
CO4: Student can be able to understand and use the basic programming constructs of C++
CO5: Students are able to learn C++ data types, memory allocation/de-allocations, functions and pointers.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔	✔	✔			
CO2	✔		✔		✔			✔		✔	✔	✔	✔
CO3	✔	✔					✔		✔	✔			✔
CO4	✔			✔	✔			✔				✔	✔
CO5	✔				✔	✔				✔	✔		

19MCAP107 - PROGRAMMING LAB - II (Data Structures using C++)

Learning Objectives:

- To write and execute programs in C++ to solve problems using data structures such as arrays, linked lists, stacks, queues, trees, graphs, hash tables and search trees.
- To write and execute write programs in C++ to implement various sorting and searching methods.
- To understand the linear data structure such as stack, queue, list etc.
- To implement the sorting and search algorithms.

Lab Exercises

1. Write a C++ programs to implement recursive and non recursive i) Linear search ii) Binary search
2. Write a C++ programs to implement i) Bubble sort ii) Selection sort iii) quick sort iv) insertion sort
3. Write a C++ programs to implement the following using an array. a) Stack ADT b) Queue ADT
4. Write a C++ programs to implement list ADT to perform following operations a) Insert an element into a list. b) Delete an element from list c) Search for a key element in list d)count number of nodes in list
5. Write C++ programs to implement the following using a singly linked list. Stack ADT b) Queue ADT
6. Write C++ programs to implement the deque (double ended queue) ADT using a doubly linked list and an array.
7. Write a C++ program to perform the following operations: a) Insert an element into a binary search tree. b) Delete an element from a binary search tree. c) Search for a key element in a binary search tree.
8. Write C++ programs for implementing the following sorting methods: Merge sort b) Heap sort
9. Write C++ programs that use recursive functions to traverse the given binary tree in a) Preorder b) in order and c) post order.
10. Write a C++ program to perform the following operations a) Insertion into a B-tree b) Deletion from a B-tree.

Course Outcomes:

CO1: Ability to identify the appropriate data structure for given problem.
CO2: Graduate able to design and analyze the time and space complexity of algorithm or program.
CO3: Ability to effectively use compilers includes library functions, debuggers and trouble shooting.
CO4: Ability to identify the dynamic memory allocation concepts.
CO5 :Implement the non-linear data structure concepts like binary tree.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔			
CO2			✔		✔				✔	✔	✔	✔	✔
CO3							✔			✔			✔
CO4	✔	✔			✔				✔	✔	✔	✔	
CO5		✔	✔			✔		✔		✔	✔		

19MCAC201: ACCOUNTING AND FINANCIAL MANAGEMENT

Learning Objectives:

- To gain fundamental knowledge on accounting and accounting process.
- To have practical knowledge in the preparation of final accounts of a concern
- To impart basic knowledge on the Tools of financial statement analysis
- To give insight into the concepts cost accounting and elements of cost
- To impart basic knowledge in the preparation of budgeting
- To give conceptual knowledge on the management of funds

Unit-I

Basics of Accounting: Meaning and Definition of Accounting – Accounting Systems-Double Entry Bookkeeping- Journal - Rules of Journalizing - Subsidiary Books- Ledger – Trial balance-Bank Reconciliation statement.

Unit-II

Income Statement and Balance Sheet: Preparation of Final Accounts of Trading Concerns – Trading Accounts – Profit and Loss Account and Balance Sheet - simple adjustments – Limitations of Financial Statements.

Unit-III

Analysis and Interpretation of Financial Statements: Need for Financial Statements Analysis-Comparative Statements–Common size Statements- Trend analysis-Ratio analysis- Liquidity, Profitability and Solvency Ratios.

Unit-IV

Basics to Cost Accounting and Marginal costing: Introduction to costing-Advantages of cost accounting- Methods of Costing-Elements of cost–Material, Labour and Overheads- Cost Sheet.

Marginal costing- Advantages-BEP-P/V Ratio and its uses

Unit-V

Budgeting and Management of Capital: Budgeting – Budgetary Control – Objectives – Types of Budgets- Production Budget – Cash Budget – Flexible Budget

Working Capital– Nature -Concept (Theory only)

Capital budgeting– Techniques of Capital Budgeting – Payback period – Accounting Rate of Return – Discounted Cash flow Technique- Net Present Value Method.

Note: The Question paper shall consist of 60 % Simple Problems and 40 % Theory.

Text Books:

1. Periasamy.P. (2009). *A Textbook of Financial , Cost and Management Accounting.* Mumbai: Himalaya Publishing House.
2. Reddy, T., & Murthy, A. (2016). *Financial Accounting.* Chennai: Margham Publications.
3. Jain, S.P., & Narang, K. (2016). *Cost Accounting .* Ludhiana: Kalyani Publishers.
4. Pillai, R., & Bagavathi, V. (2013). *Management Accounting.* New Delhi: S.Chand & Company Ltd.
5. Gupta, S. K., Sharma, R., & Gupta, N. (2015). *Financial Management .* Ludhiana: Kalyani Publishers.

References:

1. Jain, S.P., & Narang, K. (2016). *Advanced Financial Accounting.* Ludhiana: Kalyani Publishers.
2. Khan, M. Y., & Jain, P. K. (2014). *Financial Management Text, Problems and Cases.* Chennai: McGraw Hill Education (India) Private Limited.
3. Pillai, R., & Bagavathi, V. (2010). *Cost Accounting.* New Delhi : S.Chand & Company Ltd.
4. Reddy, T., & Hari Prasad Reddy, Y. (2015). *Cost Accounting.* Chennai: Margham Publications.

Course outcomes:
CO1: Explain the accounting process and it's concepts.
CO2: Prepare the final accounts of a business.
CO3: Describe the various tools of financial statement analysis.
CO4: Classify the cost and prepare cost sheet.

CO5: Prepare budgets- capital as well as functional budgets.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔	✔	✔	
CO2			✔		✔		✔		✔	✔			✔
CO3	✔			✔			✔		✔	✔			✔
CO4	✔	✔			✔				✔			✔	
CO5		✔				✔		✔		✔	✔		

19MCAC202: JAVA PROGRAMMING

Learning Objectives:

- To understand fundamentals of concepts of java programming.
- To understand the Applet, string handling and AWT concepts.
- Gain knowledge about Swing and JDBC.
- Be familiar with understand Java bean and EJB.
- To gain the Knowledge of Servlet and JSP.

Unit-I

CoreJava: Introduction-Operators-Datatypes-Variables-Arrays-Control Statements- Methods& Classes- Inheritance-package and interface- Exception handling-Multithread programming

Unit-II

I/O-Java Applet-String handling-Networking- Event Handling- Introduction to AWT-AWT controls- Layout managers-Menus-Images-Graphics.

Unit-III

Java swing: Creating a swing Applet and Application-Programming using Panes- Pluggable Look and feel-Labels- Text fields-Buttons- Toggle Buttons- Checkboxes- Radio Buttons-View Ports-Scroll Panes-Scroll Bars-List-Combo Box-Progress bars- Menus and Toolbars-Layered Panes-Tabbed Panes- Split Panes-Layouts-Windows- Dialog Boxes- Inner frame.

JDBC: The connectivity Model-JDBC/ODBC Bridge-Java.sql package-connectivity to remote database – navigating through multiple rows retrieved from a database.

Unit-IV

JavaBeans: Application Builder tools-The bean developer kit (BDK)-JAR files- Introduction-Developing a simple bean-using bound properties-The java Beans API- Session Beans-Entity Beans-Introduction to Enterprise Java Beans(EJB)-Introduction to RMI(Remote Method Invocation):A simple client-server application using RMI.

Unit-V

Java Servlets: Servlet basic-Servlet API basic-Lifecycle of a Servlet-Running Servlet-DebuggingServlet-Thread-safeServlet-HTTPRedirects-Cookies-Introduction to Java server pages (JSP).

Text Books:

1. Herbert $Schidt$, "The Complete Reference JAVA2", TataMc-Graw Hill 5th edition, 2002.

References:

1. BalagurusamyE.,"ProgrammingwithAPerimer3eJava", TataMc-Graw Hill,2007.

2. DustinR.Callway,"InsideServlets",AddisonWesley, 1999.

3. MarkWatka"UsingJava2EnterpriseEdition",Que,1stedition, 2001.

4. SetvenHolzner,"Java2 BlackBook", CoriolisGroupBooks, 2001.

5. NPTEL: https://onlinecourses.nptel.ac.in/noc19_cs07/course

Course Outcomes:

CO1: To learn the structure and model of the Java programming language.
CO2: To gain the knowledge of java programming statement.
CO3: Develop software in the Java programming language.
CO4: To gain the knowledge of Java servlets.
CO5: Evaluate user requirements for software functionality required to decide whether the Java programming language can meet user requirements.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔		✔	
CO2			✔		✔		✔		✔				✔
CO3	✔	✔			✔		✔	✔		✔			✔
CO4	✔	✔			✔		✔		✔			✔	
CO5		✔	✔			✔		✔		✔	✔		✔

19MCAC203: RELATIONAL DATABASE MANAGEMENT SYSTEM

Learning Objectives:

- To understand the different issues involved in the design and implementation of a database system.
- To study the physical and logical database designs, database modeling, relational, hierarchical, and network models
- To understand and use data manipulation language to query, update, and manage a database.
- To develop an understanding of essential DBMS concepts such as: database security, integrity, and concurrency.
- To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

Unit-I

Introduction - Database System Applications – Purpose of Database Systems – View of Data – Database Languages – Relational Databases – Database design – Data storage and Querying – Transaction Management – Database Architecture – Data Mining and Information Retrieval – Database Users and Administrators – History of Database Systems.

Unit-II

Relational Databases - Structure of the relational databases - Database schema - Keys - Schema diagrams - Relational query languages - Relational operations - Overview of SQL query language - SQL data definition - Basic structure of SQL queries - Additional basic operations - Set operations - Null values - Aggregate functions - Nested sub queries - Modification of the

database.

Unit-III

Relational Database design - Features of Good Relational Designs - Atomic Domains and First Normal Form - Decomposition Using Functional Dependencies - Functional-Dependency Theory - Algorithms for Decomposition - Decomposition Using Multivalued Dependencies - More Normal Forms - Database-Design Process.

Unit-IV

Indexing and Hashing - Basic Concepts - Ordered Indices - B+-Tree Index Files - B+-Tree Extensions - Multiple-Key Access - Static Hashing - Dynamic Hashing - Comparison of Ordered Indexing and Hashing - Bitmap Indices - Index Definition in SQL- Measures of Query Cost - Selection Operation - Sorting - Join Operation - Other Operations - Evaluation of Expressions.

Unit-V

Distributed Databases - Introduction - Homogeneous and Heterogeneous Databases - Distributed Data Storage - Distributed Transactions - Commit Protocols - Concurrency Control in Distributed Databases - Availability - Distributed Query Processing - Heterogeneous Distributed Databases - Cloud-Based Databases - Directory Systems.

Text Book:

1. Abraham Silberschatz, Henry F. Korth and S. Sudharshan, Database System Concepts, Sixth Edition, Tata McGraw Hill, 2002.

References:

1. Bipin C. Desai, An Introduction to Database Systems, Galgotia Publications, 2002.
2. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, Third Edition, Tata McGraw Hill Higher Education, 2003.
3. Elmasri and Navathe, Fundamentals of Database Systems, Sixth Edition, Pearson Education Asia, 2010.

Course Outcomes:

Co1: Define program-data independence, data models for database systems, database schema and database instances.
Co2: Recall Relational Algebra concepts, and use it to translate queries to Relational statements and vice versa.
Co3: Identify Structure Query Language statements used in creation and manipulation of Database Identify the methodology of conceptual modeling through Entity Relationship model.
Co4: Develop an understanding of the differences between OODBMS, ORDBMS and RDBMS and the practical implications of each approach.
Co5: Analyze and design a real database application.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔		✔	
CO2		✔	✔		✔		✔		✔		✔	✔	✔
CO3	✔			✔			✔			✔			✔
CO4	✔		✔		✔				✔	✔	✔	✔	
CO5		✔	✔			✔		✔		✔	✔		✔

19MCAC204: OPERATING SYSTEMS

Learning Objectives:

- To learn the fundamentals of Operating Systems.
- To learn the mechanisms of OS to handle processes and threads and their communication.
- To learn the mechanisms involved in memory management.
- To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols.
- To know the components and management aspects of concurrency management.

Unit-I

Introduction – Operating System Structure - Operating System Operations – Protection and Security – Kernel Data Structures - Computing Environments - Open Source Operating Systems - Operating System Services – User operating system Interface – System calls – Types of system calls – System programs – Operating System design and Implementation – Operating System-debugging - Operating System-Generation – System Boot.

Unit-II

Process Management – Process Concepts – Process Scheduling – Operation on Processes – Interprocess Communication – Threads – Multicore Programming - Multithreading Models – Thread Libraries - Implicit Threading - Threading Issues – Process Synchronization – The Critical Section Problem – Peterson's Solution – CPU Scheduling – Scheduling Criteria – Scheduling Algorithms - Deadlocks – System Model – Deadlock Characterization – Methods for handling Deadlocks – Deadlock Prevention – Deadlock Avoidance – Deadlock Detection – Recovery from Deadlock.

Unit-III

Memory Management – Background – Swapping – Contiguous Memory Allocation – Paging – Structure of the Page Table – Virtual Memory Background - Demand Paging – Copy-on-Write – Page Replacement – Allocation of Frames – Thrashing – Memory- Mapped Files – Allocating Kernel Memory – Other Considerations – Operating System Examples.

Unit-IV

Storage Management – Overview of Mass Storage Structure – Disk Structure – Disk Attachment – Disk Scheduling – Disk Management – Swap Space Management- RAID Structure – Stable Storage Implementation – Tertiary Storage Structure – I/O Hardware – Application of I/O Interface – Kernal I/O Subsystem - 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 – Efficiency and Performance.

Unit-V

Case study - **Windows 7**- History - Design Principles - System Components - Terminal Services and Fast User Switching - File System - Networking - Programmer Interface - **Android -** Overview of Android Operating System - The Android ecosystem - Android Architecture - Android Versioning.

Text Book:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating Systems Concepts, John Wiley & Sons, Inc., Ninth Edition, 2004.

References:

1. Andrew S. Tanenbaum, Modern Operating Systems, Prentice Hall of India, Third Edition, 2007.
2. Deitel ,H. M., Operating Systems, Pearson Education, Third Edition, 2004.
3. https://android.googlesource.com

Course Outcomes:

CO1: Analyze the structure of operating system and basic architectural components involved in design.
CO2: Analyze and design the applications to run in parallel either using process or thread models of different operating system.
CO3: Analyze the various device and resource management techniques for timesharing and distributed systems.

CO4: Understand the Mutual exclusion, Deadlock detection and agreement protocols of Distributed operating system.
CO5: Interpret the mechanisms adopted for file sharing in distributed Applications.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔			
CO2			✔		✔		✔		✔	✔		✔	✔
CO3			✔	✔			✔			✔	✔		✔
CO4	✔				✔			✔	✔			✔	
CO5	✔		✔			✔		✔		✔	✔		

19MCAC205: SOFTWARE ENGINEERING

Learning Objectives

- To understand the role of software Engineering.
- To learn the mechanisms of developing software.
- To identify the risks n software development.
- To understand the design concepts testing methods and strategies.
- To understand and collect the requirement of software engineering.

Unit-I

Introduction: AGenericViewofProcess–ProcessModels-TheWaterfallModel-Incremental Model-Evolutionary Model-Specialized Model-The Unified Process–Agile Process – Agile Models–Software Cost Estimation–Planning –Risk Analysis–Software Project Scheduling.

Unit-II

Requirement Analysis: System Engineering Hierarchy –System Modeling–Requirements Engineering: Tasks- InitiatingThe Process-Eliciting Requirements-Developing Use Cases-Negotiating Requirements-Validating Requirements–Building the Analysis Models: Concepts

Unit-III

Software Design: Design Concepts–Design Models–Pattern Based Design–Architectural Design–Component Level Design–Component–Class Based And Conventional Components Design–User Interface–Analysis And Design

Unit-IV

Software Testing : Software Testing–Strategies: Conventional-Object Oriented–Validation Testing– Criteria–Alpha–Beta Testing- System Testing –Recovery–Security–Stress–Performance –Testing Tactics–Testing Fundamentals-Black Box–While Box–Basis Path-Control Structure

Unit-V

Software configuration and Management, and Quality Assurance: Software Configuration And Management-Features-SCM Process-Software Quality Concepts – Quality Assurance – Software Review–Technical Reviews – Formal Approach To Software Quality Assurance–Reliability–Quality Standards–Software Quality Assurance Plan.

Text Book:

1.RogerPressman.S.,"SoftwareEngineering:APractitioner'sApproach",6th Edition,McgrawHill,2005.

References:

1. Fleeger P,"SoftwareEngineering",PrenticeHall,1999.
2. CarloGhezzi, MehdiJazayari, DinoMandrioli, "Fundamentals of Software Engineering", Prentice Hall ofIndia,1991.
3. Sommerville I,"SoftwareEngineering",5[th]Edition:AddisonWesley,1996.
4. Pankaj Jalote, "An Integrated Approach to Software Engineering", Third Edition, Narosa publications, 2011.
5. Ian Sommerville, "Software engineering", Ninth Edition, Pearson Education Asia, 2010.

Course Outcomes

> CO1: To gain the knowledge of Software Engineering
> CO2: Gather and specify requirement of Software Engineering
> CO3: Learn the design concepts.
> CO4: To write the test cases.
> CO5: Apply software engineering concepts in software development.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	
CO2	✔		✔		✔			✔	✔				✔
CO3				✔			✔			✔		✔	✔
CO4	✔	✔			✔				✔				
CO5	✔	✔	✔			✔		✔		✔	✔		✔

19MCAP206- Programming Lab – III (Java Programming)

Learning objectives:

- To understand how to design, implement, test, debug, and document programs that use basic data types and computation, simple I/O, conditional and control structures, string handling and functions.
- To understand the importance of Classes & objects along with constructors, Arrays and Vectors.
- Discuss the principles of inheritance, interface and packages and demonstrate though problem analysis assignments how they relate to the design of methods, abstract classes and interfaces and packages.
- To understand importance of Multi-threading & different exception handling mechanisms.
- To learn experience of designing, implementing, testing, and debugging graphical user interfaces in Java using applet and AWT that respond to different user events.

Lab Exercises

1. Write a JAVA program to implement class and object
2. Write a JAVA program to implement command line argument.
3. Write a JAVA program to implement the bitwise operators.
4. Write a JAVA program to implement method overloading.
5. Write a JAVA program to implement packages.
6. Write a JAVA program to implement interface.
7. Write a JAVA program to implement inheritance mechanism.
8. Write a JAVA program to implement exception handling.
9. Write a JAVA program to implement user-defined exception handling.
10. Write a JAVA program to implement multithreaded programming concept.
11. Write a JAVA program to implement abstract class concept.
12. Write a JAVA program to implement RMI concept.
13. Applet using Labels, Text Fields and Buttons.
14. Library Management using JDBC concept
15. Programs using Swings Concepts

Course outcomes:

CO1: Implement Object Oriented programming concept using basic syntaxes of control Structures, strings and function for developing skills of logic building activity.
CO2: Identify classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem
CO3: Demonstrates how to achieve reusability using inheritance, interfaces and packages and describes faster application development can be achieved.
CO4: Demonstrate understanding and use of different exception handling mechanisms and concept of multithreading for robust faster and efficient application development.
CO5: Identify and describe common abstract.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔			✔
CO2	✔		✔		✔		✔		✔		✔	✔	
CO3			✔		✔		✔			✔			✔
CO4	✔				✔				✔			✔	
CO5		✔	✔			✔		✔		✔	✔		✔

19MCAP207- Programming Lab - IV (RDBMS)

Learning Objectives:

- Keep abreast of current developments to continue their own professional development.
- To engage themselves in lifelong learning of Database management systems theories and technologies this enables them to purse higher studies.
- To interact professionally with colleagues or clients located abroad and the ability to overcome challenges that arises from geographic distance, cultural differences, and multiple languages in the context of computing.
- Develop team spirit, effective work habits, and professional attitude in written and oral forms, towards the development of database applications.

List of Exercises

Cycle – I (Simple SQL)

1. Employee Management System Using SQL Commands.
2. Students Management System Using SQL Commands.
3. Bank Management System Using SQL Commands.
4. Index Creation.
5. Implementation of SQL queries for route database.
6. Implementation of SQL queries for route database - part I.
7. Implementation of SQL queries for route database - Part II.
8. Creating view using SQL commands.
9. Creation of Table Partition.
10. Default trigger procedure and drop command
11. Report creation.

Cycle – II (PL/SQL)

12. Factorial of number
13. Checking whether a number is prime or not

14. Fibonacci series
15. Reversing the string
16. Swapping of two numbers
17. Odd or even number
18. Duplication of records

Course Outcomes

CO1: In drawing the ER, EER, and UML Diagrams.
CO2: In analyzing the business requirements and producing a viable model for the implementation of the database.
CO3: In converting the entity-relationship diagrams into relational tables.
CO4: To develop appropriate Databases to a given problem that integrates ethical, social, legal, and economic concerns.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	
CO2	✔		✔		✔				✔			✔	✔
CO3			✔				✔			✔			✔
CO4	✔	✔			✔			✔			✔	✔	

19MCAC301: C# and .NET FRAMEWORK

Learning objectives:

- To build deep understanding of C# language features.

- To construct strong concepts of OOP's and implement the same in C#.

- To create and manage strings, arrays, collections and enumerators using .NET framework library and perform file input-output, network, multi threading operations.

- To develop database centric applications using ADO.NET.

- To build GUI and web-based applications using .NET Framework.

Unit-I

The .NET framework: Introduction – Architecture of .NET framework - Common Language Runtime - Common type system- Common language specification- Base class library - Intermediate language - Just-in-time compilation – Managed code vs. unmanaged code – Assemblies - interoperability - garbage collection – Application Domain - Security.

Unit-II

C# Basics: Introduction- Data types - Identifiers- Variable & Constants- C# statements - Classes and Objects- Arrays and Strings- Methods and Classes – Operator overloading – Indexers and Properties – Inheritance – Interfaces - Delegates and Events

Unit-III

C# Using Libraries: Namespace- Collections - Using I/O - Multithreaded programming: Fundamentals –Thread class - Thread Priorities – Synchronization; Exception Handling - Networking through internet : System.Net members - Internet access fundamentals- Handling network errors.

Unit-IV

Advanced   Features Using C#: Creating form based windows applications – Building Components - Data access with .NET : Overview of ADO.NET – using Database Connections – Commands – Dataset class –XML Schemas - Populating a Dataset – Persisting Dataset changes -Working with ADO.NET

Unit-V

Web programming: ASP.NET introduction – Architecture - ASP.NET server controls – ADO.NET and Data Binding – Web Services

Text Books:

1. Hebert Schildt, "C# 2.0: The Complete Reference", Tata Mc-Graw Hill, 2006.

2. Simon Robinson, Christian Nagel, Karli Watson, Jay Glynn, Morgan Skinner, Bill Evjen, "Professional C#" 3rd edition, Wiley Publications, 2004.

Reference Books:

1. Jeffrey Richter, "Applied Microsoft Net Framework Programming", Microsoft Press, 2002.

2. Fergal Grimes, "Microsoft Net for Programmers",.Manning Publication, 2002.

3. Tony Baer, Jan D. Narkiewicz, Kent Tegels, Chandu Thota, Neil Whitlow, "Understanding the Net Framework", Wrox Press, 2002.

4. Balagurusamy, "Programming with C#", TataMc-Graw Hill, 2002.

5. Shibi Panikkar and Kumar Sanjeev, "Magic of C# with NET Frame Work", Firewall Media,. 2005.

Course outcomes:

CO1: Recognize, diagram, and implement introductory programming concepts using C#
CO2: Determine logical alternatives with C# decision structures utilizing iteration, class methods, fields, and properties.
CO3: Assemble forms, classes, and controls into C# solutions utilizing arrays and file/database access methods.
CO4: Develop windows applications.
CO5: Develop web-based applications.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✓	✓		✓		✓		✓		✓			
CO2			✓		✓				✓	✓	✓	✓	✓
CO3							✓			✓			✓
CO4	✓	✓			✓				✓	✓	✓	✓	
CO5		✓	✓			✓		✓		✓	✓		

19MCAC302: COMPUTER NETWORKS

Learning Objectives:

- Study the basic taxonomy and terminology of the computer networking and enumerate the layers of OSI model and TCP/IP model.
- Acquire knowledge of Application layer and Presentation layer paradigms and protocols.
- Study Session layer design issues, Transport layer services, and protocols.
- Gain core knowledge of Network layer routing protocols and IP addressing.
- Study data link layer concepts, design issues, and protocols.

Unit-I

Introduction: The uses of computer networks-Network hardware-Network software-Reference models-Example of networks-Network standardization.

The physical layer: The theoretical basis for data communication–Guided Transmission media – Wireless transmission– PSTN-Mobile telephone-Communication satellite.

Unit-II

The Data Link Layer : Data link layer design issues-Error detection and correction – Elementary data link protocols- Sliding window protocols- Example of data link protocols-ETHERNET–802.11-802.16-Bluetooth-Data link layer Switching.

Unit-III

The network layer: Network layer design issues- Routing algorithms- Congestion control algorithms-Internetworking-Network layer in Internet. Network Services BOOTP and DHCP-Domain Name Service-WINS-Web Serving and Surfing Web servers-Web clients (browsers).

Unit-IV

The transport layer: Transport layer design issues-Transport protocols-Simple transport protocol-Internet transport protocols UDP-TCP.

Unit-V

The application layer: Domain name system- Electronic mail- World wide web– Multimedia–Cryptography-Digital signature-Communication Security.

Text Book:

1.Andrew S.Tanenbaum, "Computer networks", PHI, 4thedition,2002.

References:

1. William Stallings," Data and computercommunications",PHI-2001.
2. DouglasE. comer,"Internet working with TCP/IP-Volume-I",PHI,1997.
3. Larry L. Peterson and Bruce S. Davie, "Computer Networks: A Systems Approach", Fifth Edition, Morgan Kaufmann Publishers, 2012.
4. William Stallings, "Data and Computer Communications", Tenth Edition, Pearson, 2013.
5. James F. Kurose, Keith W. Ross, "Computer Networking: A Top-DownApproach", Fifth Edition, Pearson Education,2012

Course Outcomes:

CO1: To master the terminology and concepts of the OSI reference model and the TCP- IP reference model.

CO2: To master the concepts of protocols, network interfaces, and   design/performance issues in local area networks and wide area networks.

CO3: To be familiar with wireless networking concepts.

CO4: To be familiar with contemporary issues in networking technologies.

CO5: To be familiar with network tools and network programming.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✓			✓		✓		✓		✓		✓	
CO2	✓		✓	✓	✓				✓		✓	✓	✓
CO3		✓				✓	✓			✓	✓		✓
CO4	✓		✓		✓				✓			✓	
CO5		✓				✓		✓			✓		✓

19MCAC304: COMPUTER GRAPHICS AND MULTIMEDIA

Learning Objectives:

- Gain knowledge about graphics system.
- Understand the two dimensional graphics and their transformations.
- Realize illumination and color models.
- Understand the three dimensional graphics and their transformations.
- Be familiar with understand various clipping techniques.

Unit-I

Introduction: Overview of Graphics System - Line Drawing Algorithms and Circle Drawing Algorithms - 2D Clipping Algorithms.

Unit-II

2D Transformations: Window view port mapping transformations – 2D Viewing – Spline curves – Bezier Curves - Polygons - Interactive Input methods.

Unit-III

3D Transformations: 3D Concepts - 3D Viewing – Visible Surface Detection Methods - Visualization and polygon rendering – Color models –XYZ-RGB-YIQ-CMY-HSV Models – Animation – Key Frame systems - General animation functions - morphing.

Unit-IV

Multimedia: Multimedia hardware & software - Components of multimedia – Multimedia Architecture – Multimedia Data base systems –multimedia Authoring Tools – Synchronization Issues – Multimedia Applications: Video conferencing – Virtual reality–Interactive video – video on demand.

Unit-V

Multimedia in Web applications: Basic web graphics, Web page design and site building, Adding multimedia to the web MACROMEDIA DREAMWEAVER: Planning, Designing (Tables, layers, templates, style sheet), Building and Publishing a web site.

CASE STUDY: Creating web site with graphics, animations, audio, video and interaction.

Text Books:

1. Hearn D and Baker M.P, "Computer Graphics", PHI, India. (Unit 1, 2 & 3)

2. Ralf Steinmetz, Klara Steinmetz,"Multimedia Computing, Communications and Applications", Pearson Education, 2004. (Unit 4)

3. Macromedia Dreamweaver Basics and Programming. (Unit 5)

References:

1. William M. Newman, Robert F.Sproull, "Principles of interactive computer graphics", II Edition, McGraw Hill, 1989.

2. Steven Harrington, "Computer Graphics A programming Approach", McGraw Hill, 1987.

3. Nian Li, Mark S.Drew, Fundamentals of Multimedia , Pearson Educn., 1st Indian Reprint 2004.

4. Tay Vaughan, Multimedia Making it Work, Tata McGraw -Hill, Seventh Edition, 2007.

5. John F. Koegel Buford, Multimedia Systems , Pearson Education, 10th Indian Reprint 2004.

Course Outcomes:

CO1: Design two dimensional graphics and apply two dimensional transformations.
CO2: Design three dimensional graphics and apply three dimensional transformations.
CO3: Apply Illumination and color models.
CO4: Apply clipping techniques to graphics.
CO5: Design an Animation sequence.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	✔
CO2			✔		✔		✔	✔	✔				✔	✔
CO3		✔	✔				✔			✔			✔
CO4	✔	✔			✔	✔	✔		✔	✔	✔	✔	
CO5		✔		✔	✔			✔		✔	✔		

19MCAC304: SOFTWARE TESTING AND QUALITY ASSURANCE

Learning Objectives:

- Develop methods and procedures for software development that can scale up for large systems.

- It can be used to consistently produce high-quality software at low cost and with a small cycle time.

- Student learns systematic approach to the development, operation, maintenance, and retirement of software.

- Student learns how to use available resources to develop software, reduce cost of software and how to maintain quality of software.

- Methods and tools of testing and maintenance of software's.

UNIT- I

Testing Environment and Test Processes: World-Class Software Testing Model – Building a Software Testing Environment - Overview of Software Testing Process – Organizing for Testing – Developing the Test Plan – Verification Testing –Analyzing and Reporting Test Results – Acceptance Testing – Operational Testing – Post Implementation Analysis

UNIT-II

Testing Techniques and Levels of testing: Using White Box Approach to Test design - Static Testing Vs. Structural Testing – Code Functional Testing – Coverage and Control Flow Graphs –Using Black Box Approaches to Test Case Design – Random Testing – Requirements based testing –Decision tables –State-based testing – Cause-effect graphing – Error guessing – Compatibility testing – Levels of Testing - Unit Testing – Integration Testing - Defect Bash Elimination. System Testing - Usability and Accessibility Testing – Configuration Testing - Compatibility Testing - Case study for White box testing and Black box testing Techniques.

UNIT- III

Incorporating Specialized Testing Responsibilities: Testing Client/Server Systems – Rapid Application Development Testing – Testing in a Multiplatform Environment – Testing Software System Security - Testing Object-Oriented Software – Object Oriented Testing – Testing Web based systems – Web based system – Web Technology Evolution – Traditional Software and Web based Software – Challenges in Testing for Web-based Software – Testing a Data Warehouse - Case Study for Web Application Testing.

UNIT-IV

Test Automation: Selecting and Installing Software Testing Tools - Software Test Automation – Skills needed for Automation – Scope of Automation – Design and Architecture for Automation – Requirements for a Test Tool – Challenges in Automation – Tracking the Bug – Debugging – Case study using Bug Tracking Tool.

UNIT-V

Software Testing and Quality Metrics: Testing Software System Security - Six-Sigma – TQM - Complexity Metrics and Models – Quality Management Metrics - Availability Metrics - Defect Removal Effectiveness - FMEA - Quality Function Deployment – Taguchi Quality Loss Function – Cost of Quality. Case Study for Complexity and Object Oriented Metrics.

Text Books:

1. William Perry, "Effective Methods of Software Testing", Third Edition, Wiley Publishing 2007.

2. Srinivasan Desikan and Gopalaswamy Ramesh, "Software Testing – Principles and Practices", Pearson Education, 2007.

3. Naresh Chauhan , "Software Testing Principles and Practices " Oxford University Press , New Delhi , 2010.

4. Dale H. Besterfiled et al., "Total Quality Management", Pearson Education Asia, Third Edition, Indian Reprint (2006).

5. Stephen Kan, "Metrics and Models in Software Quality", Addison – Wesley, Second Edition, 2004.

References:

1. Llene Burnstein, " Practical Software Testing", Springer International Edition, Chennai, 2003

2. Renu Rajani,Pradeep Oak, "Software Testing – Effective Methods, Tools and Techniques", Tata McGraw Hill,2004.

3. Edward Kit, "Software Testing in the Real World – Improving the Process", Pearson Education,1995.

4. Boris Beizer, "Software Testing Techniques" – 2nd Edition, Van Nostrand Reinhold, New York,1990.

5. Adithya P. Mathur, "Foundations of Software Testing – Fundamentals algorithms and

   Techniques", Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.

Course Outcomes:

CO1: Apply modern software testing processes in relation to software development and project management.

CO2: Create test strategies and plans, design test cases, prioritize and execute them.

CO3: Manage incidents and risks within a project.

CO4: Contribute to efficient delivery of software solutions and implement improvements in the software development processes.

CO5: To gain expertise in designing, implementation and development of computer based systems and IT processes.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔		✔	
CO2	✔		✔		✔			✔	✔			✔	✔
CO3		✔		✔			✔			✔	✔		✔
CO4	✔	✔			✔		✔		✔			✔	
CO5		✔	✔			✔		✔		✔	✔		✔

19SMATX04: RESOURCE MANAGEMENT TECHNIQUES

Learning Objectives:

- Resource management techniques find applications in diverse fields including Engineering, Management Science, Computer Science and Economics.
- In this course, the general linear programming problem, simplex computation procedure, revised simplex method, duality problems in linear programming and some nonlinear programming problems, Integer programming problem, transportation and assignment problems, PERT and CPM are also covered .
- The main objective is to solve varieties of problems.

Unit-I

Linear programming(LP)LP formulation and graphical solution –the simplex method- revised simplex method.

Unit-II

Duality and networks-definition of the dual problem-primal-Dual relationships-Dual simplex method -transportation and assignment models - transshipment models – network minimization-shortest route problems.

Unit-III

Integer programming - cutting plane algorithms, Branch and bound Algorithm - Multistage(dynamic) programming solution of LP by dynamic programming.

Unit-IV

Classical optimization theory: unconstrained problem – Jacobian method-Lagrangian method- khuntucker conditions- simple problems.

Unit-V

Project scheduling- network diagram representation –critical path Computation-time charts and resources levelling– PERT Networks

Text Book:

1. TahaA.H.,operationsresearchanintroduction,macmillanpublishingcompany, Newyork,1997.

References:

1. Billey E. Gillet, "Introduction To Operations Research A Computer Oriented Algorithmic Approach", TataMcGrawHill, NewDelhi,1979.
2. Hamdy A. Taha, A.M. Natarajan, P.Balasubramnie and A.Tamilarasi, "Operations Research- An Introduction", 2009.
3. Kandi swarup, P.K Gupta and Manmohan, "Operations Research", 18the edition, 2015.
4. Ravidran, Phillips and Solterg, "Operations Research: Principles & Practice", 2nd EDC(WSE series),2007.

Course outcomes:

CO1: Model any real life situation into a mathematical model.

CO2: Solve the problem for the required demand.

CO3: Optimize the transportation and assignment of jobs.

CO4: Upgrade their ability in production management through project scheduling and allocation of   resources.

CO5: Develop their personnel management through manpower planning and salary administration.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔			✔
CO2	✔		✔		✔		✔		✔	✔		✔	
CO3			✔			✔	✔			✔	✔		✔
CO4	✔				✔				✔				
CO5		✔	✔			✔		✔				✔	✔

19MCAP305 - Programming Lab- V (Software Testing)

Learning Objectives:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an as yet undiscovered error.
- A successful test is one that uncovers an as yet undiscovered error.
- Documenting user requirements using the UML notation.
- Description of the various types of the Use Cases.

List of Exercises:

1. Understand The Automation Testing Approach
2. Using Selenium IDE, Write a test suite containing minimum 4 test cases
3. Write and test a program to login a specific web page
4. Write the test cases for any known application (e.g. Banking application)
5. Create a test plan document for any application (e.g. Library Management System)
6. Study of any testing tool (e.g. Win runner)
7. Study of any web testing tool (e.g. Selenium)
8. Study of any bug tracking tool (e.g. Bugzilla, bugbit)
9. Study of any test management tool (e.g. Test Director)
10. Study of any open source-testing tool (e.g. Test Link)

Course Outcomes:

CO1: Apply modern software testing processes in relation to software development and project management.

CO2: Create test strategies and plans, design test cases, prioritize and execute them.

CO3: Manage incidents and risks within a project.

CO4: Contribute to efficient delivery of software solutions and implement improvements in the software development processes.

CO5: To gain expertise in designing, implementation and development of computer based systems and IT processes.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔			✔
CO2		✔	✔		✔		✔		✔			✔	
CO3	✔			✔			✔			✔			✔
CO4		✔			✔		✔		✔		✔	✔	
CO5	✔	✔				✔		✔		✔	✔		

19MCAP307 - Programming Lab V I (C# and .Net Programming)

Learning Objectives

- To impart basic knowledge of different control statements and array associated with C # programming.
- To learn various C# elements and OOPS concepts.
- To learn interface, delegates, event and error handling concepts in C#.
- To impart knowledge on networking including socket programming and reflection.
- To acquire a working knowledge of windows and web based applications.

List of Exercises

1. Finding Prime number using Classes and Objects

2. Separating Odd/Even Number into Different Arrays

3. String Manipulations

4. Jagged Array manipulation

5. Implementing 'ref' and 'out' keywords

6. Implementing 'Params ' keyword

7. Boxing and Unboxing

8. Constructor Overloading

9. Implementing property

10. Implementing indexer

11. Implementing Multiple inheritance using Interface

12. Implementing Abstract Class

13. Exception Handling Using Try, Catch, and Finally

14. Demonstrating multicast Delegates

15. Implementing the Concept of Reflection

16. Socket Programming

17. Simple Calculator-A Window Application

18. Student Profile-A Window Application

19. Palindrome-A Web Application

20. Formatting Text-A Web Application

Course Outcomes:

On successful completion of the course, the students will be able to:

CO1: Develop correct, well-documented C# programs using control statements.
CO2: Develop object oriented programming using C# classes and objects.
CO3: Handle the exception and event-driven programs.
CO4: Perform network based programming including chat applications.
CO5: Develop windows and web based applications

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔		✔	
CO2		✔	✔		✔				✔		✔		✔
CO3				✔			✔			✔			✔
CO4	✔	✔			✔		✔		✔			✔	
CO5	✔		✔			✔		✔		✔	✔		

19MCAC401: COMPILER DESIGN

Learning Objectives:

- To introduce the major concept areas of language translation and compiler design.
- To enrich the knowledge in various phases of compiler ant its use, code optimization techniques, machine code generation, and use of symbol table.
- To extend the knowledge of error recovery, code generation, and code optimization.
- To extend the knowledge of parser by parsing LL parser and LR parser.
- To provide practical programming skills necessary for constructing a compiler.

Unit-I

Introduction To Compilers: Translators-Compilation and Interpretation- The phases of Compiler-Errors encountered in different phases-The grouping of phases- Compiler construction tools-A simple one-pass compiler – Language design-Programming language grammars-Derivation-Reduction and Ambiguity.

Unit-II

Lexical Analysis: Need and role of lexical analyzer-Input Buffering-Lexical errors-Expressing tokens by Regular Expression –Finite Automata-Converting regular expression to NFA-Converting NFA to DFA-Minimization of DFA-Language for specifying lexical analyzers-LEX-Design of lexical analyzer for a sample language.

Unit-III

Syntax Analysis: Need and role of the parser-Context Free Grammars-Top Down parsing-Recursive Parsing-Problems- Recursive Descent parser- Predictive Parser – LL(1)Parser-Bottom

up parsers-shift reduce parser-operator precedence parsers-LR parser–LR(0)item–Construction of SLR Parsing table–CLR parser–LALR Parser. Error handling and recovery in syntax analyzer-YACC-Design of a syntax analyzer for a sample language.

Unit-IV

Syntax Directed Translation: Syntax-directed definitions-Construction of syntax trees-Bottom-up evaluation, L-attributed definitions-Top down translation, Recursive Evaluator Method, Comparison of Translation Methods. Syntax directed translation for declaration statements, assignment statements, Boolean expression, control flow statements, procedure calls.

Unit-V

Run-Time Environment: Source language issues-Storage organization-Storage allocation-access to non local names-parameter passing-Symbol tables.

Code Optimization and Code Generation: Principal sources of Optimization-Optimization of basic blocks-Global Optimization-Global dataflow analysis-Efficient data flow algorithms-Issues in design of a code generator-A simple code generator algorithm.

Text Book:

1. AlfredVAho, RaviSethiand JeffreyD Ullman,"Compilers– Principles,Techniques andTools", PearsonEducation,NewDelhi,2004.

References:

1. Sudha Sadasivam G, "Compiler Design", Scitech Publications (India) Private Limited, Chennai,2008.

2. DickGrone,Henri E Bal,Ceriel JH Jacobs and KoenG Langendoen,"Modern Compiler Design",John Wiley &Sons, USA, 2000.

3. DhamdhereDM,"Compiler Construction Principles & Practice", Macmillan   India Limited, NewDelhi, Second Edition,1997.

4. Jean Paul Tremblay and Paul G Serenson,"The Theory & Practice of Compiler Writing", McGrawHill Publishing Company,NewDelhi,1985.

Course Outcomes:

CO1: To apply the knowledge of lex tool & YACC tool to develop a scanner & parser.

CO2: To design & conduct experiments for Intermediate Code Generation in compiler.

CO3: To design & implement a software system for backend of the compiler.

CO4: To learn the new code optimization techniques to improve the performance of a program in terms of speed & space.

CO5: To acquire the knowledge of modern compiler & its features.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✓	✓		✓		✓		✓			✓	✓	
CO2	✓		✓		✓			✓	✓		✓	✓	✓
CO3				✓			✓			✓			✓
CO4	✓		✓		✓				✓			✓	
CO5			✓			✓		✓		✓		✓	

19MCAC402: WEB TECHNOLOGY

Learning Objectives

- To learn about Java, HTML, DHTML concepts.
- Deploy Java Applets and Servlets.
- To know about appropriate client-side or Server-side applications.
- To gain the Knowledge of XML and its applications.
- To know about PHP scripts and create adaptive web pages.

Unit-I

HTML: History of the Internet and World Wide Web – HTTP, SMTP, POP3, MIME, Understanding roles of Web Browsers and Web Servers. Structure of HTML, Text formatting, Text styles, hyper link, image, and tables.

Unit-II

Frames, Forms and CSS: Frames, Forms and controls, Embedding audio, video and animated files in HTML, CSS –Understanding CSS, Internal CSS, External CSS, Font Properties, Text Properties, Color and Background properties, Table properties, Numbering and List Properties.

Unit-III

JavaScript: Data types and literals, operators, conditional statements, loop constructs, reserved words; core Objects: Array Object, Date Object; Functions: passing value to JavaScript functions, user defined functions, Handling old browsers , java script events, formatting cookie, retrieving cookie value from the cookie file, removing a cookie , animations using events.

Unit-IV

ASP & XML: ASP – Working of ASP – Request and Response Objects – File System Objects – Session tracking and cookies – ADO – Access a Database from ASP – XML – Structure in Data – Name spaces – DTD – Vocabularies – Accessing Web servers – IIS – Apache web server.

Unit-V

PHP & MySQL: Why PHP and MySQL - Server-Side Web Scripting - Getting Started with PHP - Adding PHP to HTML -Syntax and Variables - Control and Functions - Passing Information between Pages – Strings – Arrays and Array Functions – Numbers - MySQL Database Administration - PHP/MySQL Functions -Displaying Queries in Tables - Building Forms from Queries.

Text Books

1. Kris Jamsa, konrad King and Andy Anderson, "HTML & Web Design Tips and Techniques", Tata McGraw-Hill, First Edition, 2002.

2. Powell T.A, HTML: Complete Reference, Tata McGraw-Hill, Fifth Edition, 2010.

3. Deitel & Deitel, Goldberg, Internet and World Wide Web – How to Program, Third Edition, Pearson Education Asia, 2005.

4. Elizabeth Naramore, Jason Gerner, Yann Le Scouarnec, Jeremy Stolz, and Michael K. Glass, "Beginning PHP5, Apache, and MySQL Web Development", First Edition, Wrox publications, 2005.

5. Rajkamal, "Web Technology", First Edition, Tata McGraw-Hill, 2001.

6. Tim Converse, Joyce Park and Clark Morgan "PHP5 and MySQL Bible", Wiley Publishing, Inc,2004.

Course Outcomes

CO1: Develop a dynamic webpage by the use of java script and DHTML.

CO2: Write a well formed / valid XML document.

CO3: Connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.

CO4: Write a server side java application called Servlet to catch form data sent from client, process it and store it on database.

CO5: Write a server side java application called JSP to catch form data sent from client and store it on database.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	
CO2	✔		✔		✔		✔		✔		✔	✔	✔
CO3		✔		✔			✔			✔			✔
CO4	✔		✔		✔				✔		✔	✔	
CO5		✔				✔		✔		✔		✔	

19MCAC403: MOBILE COMPUTING

Learning Objectives:

- To learn about the concepts and principles of Mobile computing.
- To explore theoretical issues of Mobile computing.
- To develop skills of finding solutions and building software for Mobile computing applications.
- To study the specifications and functionalities of various protocols/standards of mobile networks.
- To learn Android and IOS platform and its architecture.

UNIT-I

Introduction: Aspects of Mobility – Mobile Device Profiles – Device Portability – Mobile Applications – Characteristics and Benefits – Application Model – Infrastructure and Managing Resources – Frameworks and Tools – Generic UI Development – Visual UI – Text to Speech Techniques – Multimodal and Multichannel UI .

UNIT-II

Tools: Google Android Platform – Eclipse Simulator – Android Application Architecture – Event based programming – Apple iPhone Platform – UI and Toolkit Interfaces – Event handling – Graphical Services – Animation Techniques.

UNIT-III

Application Design: Memory Management – Design Patterns for Limited Memory - Work Flow for Application development – Techniques for Composing Applications - Dynamic Linking - Plug ins and rule of thumb for using DLLs - Concurrency and Resource Management - Look and Feel

UNIT-IV

Application Development: Intents and Services – Storing and Retrieving data – Communication via the Web – Notification and Alarms – Graphics and Multimedia – Telephony – Location based Services – Packaging and Deployment – Security and Hacking

UNIT-V

Cellular Networks And Wireless LANs: Cellular Network Structure and Operation - Principles - Tessellation, Frequency Reuse, Hand off - GSM - System Architecture, Elements, Interfaces, Frame Structure, Protocol Stack, Types of Handover - IEEE 802.11 WLAN - Architecture, Reference Model - Physical layer - MAC Layer - CSMA/CA- Interference Spacing – Security – WEP,802.1x Authentication.

Text Books:

1. Reza B'Far, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", Cambridge Press University, 2009.
2. Asoke K Talukder, Hasan Ahmed, Roopa R Yavagal, "Mobile Computing Technology, Applications and Service Creation", 2nd ed, Tata McGraw Hill, 2010.

References:
1. Reto Meier, "Professional Android 2 Application Development", Wrox Wiley, 2010.
2. Neil Smyth, "iPhone iOS 4 Development Essentials – Xcode", 4 Edition, Payload media, 2011.
3. Ben Shneiderman and Catherine Plaisant, "Designing the User Interface: strategies for Effective Human Computer Interaction", Addison– Wesley, 5th Edition, 2009.
4. Zigurd Mednieks, Laird Dornin, G, Blake Meike and Masumi Nakamura, "Programming Android", O"Reilly, 2011.
5. Reto Meier, Wrox Wiley, "Professional Android 2 Application Development", 2010.
6. Alasdair Allan, "iPhone Programming", O"Reilly, 2010.
7. Wei-Meng Lee, "Beginning iPhone SDK Programming with Objective-C", Wrox Wiley, 2010.
8. Stefan Poslad, "Ubiquitous Computing: Smart Devices, Environments and interactions", Wiley, 2009.

Course Outcomes:

CO1: Grasp the concepts and features of mobile computing technologies and applications.

CO2: Have a good understanding of how the underlying wireless and mobile communication networks work, their technical features, and what kinds of applications they can support.

CO3: Identify the important issues of developing mobile computing systems and applications.

CO4: Develop mobile computing applications by analyzing their characteristics and requirements, selecting the appropriate computing models and software architectures, and applying standard programming languages and tools.

CO5: Describe Android platform, Architecture and features and design User Interface and develop Android App.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	
CO2	✔		✔		✔		✔		✔		✔		✔
CO3			✔	✔			✔	✔		✔			✔
CO4	✔	✔			✔				✔			✔	
CO5		✔	✔			✔		✔		✔	✔		✔

19MCAP406 - Programming lab-VII (Web Programming)

Learning Objectives

- To understand the concept of web technologies.
- To creating web pages by using HTML Tags.
- To understand the importance of CSS in creating a web application.
- To understand the importance of Java Script in creating a web Application
- To understand the use of XML in Document type Definition.
- To know about PHP scripts and create adaptive web pages.

List of Exercises

1. Write a HTML Program to using Image, Link and Formatting tags.

2. Write a HTML Program to using table tag of your class Time table.

3. Write a Forms in Html

4. Write a HTML program to illustrate Frame tag.

5. Forms in CSS

6. Write a program to Document Type Definition in XML.

7. Write a program Form Validation using JavaScript.

8. Write a Calculator program in Java script.

9. Write a program Multiplication table using Java script.

10.Connection in My sql with php

11.Insert record in mysql with php

12.Create,Insert,Delete,Edit in mysql with php

Course Outcome:

CO1: Develop to build a complete website using HTML
CO2: Create web pages using DHTML and Cascading Style Sheets.

CO3: Able to include JavaScript for form validations and email validations.
CO4: Write a DTD XML document.

CO5: Develop a simple web application using server side PHP programming and Database Connectivity using MySQL.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔	✔		✔
CO2	✔		✔		✔		✔			✔		✔	✔
CO3			✔				✔		✔	✔			✔
CO4	✔		✔		✔			✔			✔	✔	
CO5	✔			✔		✔		✔		✔	✔	✔	

19MCAP407 - Programming lab-VIII (MOBILE APPLICATION DEVELOPMENT)

Learning Objectives:

- To introduce Android platform and its architecture.
- To learn activity creation and Android UI designing.
- To be familiarized with Intent, Broadcast receivers and Internet services.
- To work with SQLite Database and content providers.
- To integrate multimedia, camera and Location based services in Android Application.

List of Exercises

1. Develop an application that uses GUI components, Font and Colors.

2. Implement an android application that demonstrates the use of Button, Text view & Edit text.
3. Implement an android application that demonstrates the use of Radio button, Radio group and Checkbox.
4. Develop a native calculator application.
5. Implement an android application that demonstrates the use of Intents.
6. Develop an application that uses Layout Managers and Event Listeners.
7. Develop a native application that uses GPS location information.
8. Develop an application that makes use of database.
9. Implement an application that writes data to the SD card.
10. Implement an application that creates an alert upon receiving a message.

Course Outcomes:

CO1: Understand Android platform, Architecture and features.

CO2: Design User Interface and develop activity for Android App.

CO3: Use Intent, Broadcast receivers and Internet services in Android App.

CO4: Design and implement Database Application and Content providers.

CO5: Use multimedia, and Location based services in Android App.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✓	✓		✓		✓		✓		✓		✓	
CO2			✓		✓		✓		✓		✓	✓	✓
CO3	✓		✓				✓			✓			✓
CO4	✓	✓		✓	✓				✓			✓	
CO5		✓	✓			✓		✓			✓	✓	

19MCAC501: BIG DATA ANALYTICS

Learning Objectives:

- To understand the concepts of Big Data Analytics and gain ability to design high scalable systems.
- To understand various statistical models.
- To understand the frequent item set and clustering concept.
- To understand big data and use cases from selected business domains.
- To learn, Install, configure, and run Hadoop and HDFS.

Unit-I

Introduction to Big Data: Introduction to Big Data Platform – Traits of Big data -Challenges of Conventional Systems - Web Data – Evolution Of Analytic Scalability - Analytic Processes and Tools - Analysis vs Reporting - Modern Data Analytic Tools - Statistical Concepts: Sampling Distributions - Re-Sampling - Statistical Inference - Prediction Error.

Unit-II

Data Analysis: Regression Modeling - Multivariate Analysis - Bayesian Modeling - Inference and Bayesian Networks - Support Vector and Kernel Methods - Analysis of Time Series: Linear Systems Analysis - Nonlinear Dynamics - Rule Induction - Neural Networks: Learning And Generalization - Competitive Learning - Principal Component Analysis and Neural Networks - Fuzzy Logic: Extracting Fuzzy Models from Data - Fuzzy Decision Trees - Stochastic Search Methods.

Unit-III

Mining Data Streams: Introduction To Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window - Real time Analytics Platform(RTAP) Applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

Unit-IV

Frequent Item sets and Clustering: Mining Frequent Item sets - Market Based Model – Apriori Algorithm – Handling Large Data Sets in Main Memory – Limited Pass Algorithm – Counting Frequent Item sets in a Stream – Clustering Techniques – Hierarchical – K-Means – Clustering High Dimensional Data – CLIQUE And PROCLUS – Frequent Pattern based Clustering Methods – Clustering in Non-Euclidean Space – Clustering for Streams and Parallelism.

Unit-V

Hadoop and R for Visualization: Background and fundamentals-moving data in and out of Hadoop-data serialization-applying Map Reduce patterns to big data- streaming big data-integrating R and Hadoop for statistics and more-predictive analytics with Mahout- Hacking with Hive-Programming pipelines with pig – HBase-MySQL-NoSQL- RHadoop

Text Books:

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.

2. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.

References:

1. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley & sons, 2012.

2. Glenn J. Myatt, "Making Sense of Data", John Wiley & Sons, 2007.

3. Pete Warden, "Big Data Glossary", O'Reilly, 2011.

4. Jiawei Han, MichelineKamber, "Data Mining Concepts and Techniques", Second Edition, Elsevier, second edition, 2006.

5. Alex Holmes, "Hadoop in Pracice",2012 by Manning Publications,2012.

6. Ohri A, "R for Business Analytics", Springer, 2012.

7. Prabhanjan Narayanachar Tattar, "R Statistical Application Development by Example Beginner's Guide" , packt publishing,2013.

Course Outcomes:

CO1: Explain the concepts of big data analysis.
CO2: Identify the various Big data management, processing techniques
CO3: Explain NoSQL big data management.
CO4: Analyze performance of big data analysis in Hadoop environment.
CO5: Use Hadoop related tools such as HBase, Cassandra, Pig, and Hive for big data analytics.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	
CO2	✔		✔		✔		✔		✔		✔	✔	✔
CO3				✔			✔			✔			✔
CO4		✔			✔		✔		✔		✔	✔	
CO5	✔	✔	✔			✔		✔		✔		✔	

19MCAC502: PYTHON AND R PROGRAMMING

Learning Objectives:

- To acquire knowledge in Python programming.
- To acquire knowledge in R programming.
- To develop Python programs with conditionals and loops and data structures.
- To learn how to design and program Python applications.
- To learn how to build and package Python modules for reusability.

UNIT-I

Introduction to Python Programming: Python interpreter and interactive mode; values and types variables, expressions, statements, tuple assignment, Order of operations, comments, debugging; modules and functions: function Calls, adding new functions, Definitions and Uses, flow of execution, parameters and arguments, Fruitful functions. Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, range, break, continue, pass; recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays.

UNIT-II

Lists, Tuples, Dictionaries: Lists: Traversing a List, list operations, list slices, list methods, Map, Filter and Reduce, list loop, mutability, aliasing, cloning lists, list parameters; Dictionaries: operations and methods; advanced list processing - list comprehension; Tuples: tuple assignment, tuple as return value.

UNIT-III

Files, Modules, Packages: Files and exception: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages: PANDAS, NUMPY, SCIKIT-LEARN;

UNIT-IV

Introduction to R Programming: Introduction and Preliminaries, numbers and vectors, Objects, their modes and attributes, Ordered and unordered factors, Arrays and matrices, Lists and data frames, Grouping, loops and conditional execution, functions.

UNIT-V

Statistical Models, Graphical Procedures, Packages: Statistical models : Defining statistical models; formulae, Linear models, Generic functions for extracting model information, Analysis of variance and model comparison, Updating fitted models, Generalized linear models, Nonlinear least squares and maximum likelihood models; Graphical procedures: High-level and low-level plotting commands, graphics parameters and list, Dynamic graphics. Packages: Standard packages, Contributed packages and CRAN, Namespaces.

Text Books:

1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist", 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016.

2. Guido van Rossum and Fred L. Drake Jr, ―An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

3. William N. Venables, David M. Smith, An Introduction to R: A Programming Environment for Data Analysis and Graphics, 2nd edition, Network Theory Ltd, 2009.

4. John V Guttag, ―Introduction to Computation and Programming Using Python", Revised and expanded Edition, MIT Press , 2013.

Course outcomes:

CO1: Problem solving and programming capability.
CO2: Construct and execute basic programs in Python.
CO3: Use external libraries and packages with Python.
CO4: Construct and execute basic programs in R using elementary programming techniques.
CO5: Use external R-packages in statistics and graphics.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔	✔	✔	
CO2			✔		✔				✔			✔	✔
CO3		✔		✔			✔			✔	✔		✔
CO4	✔				✔		✔		✔			✔	
CO5		✔	✔	✔		✔		✔		✔	✔		

19MCAC503: CLOUD COMPUTING

Learning objectives:

- To learn how to use Cloud Services.
- To implement Virtualization
- To implement Task Scheduling algorithms.
- Apply Map-Reduce concept to applications.
- To build Private Cloud.

Unit-I

Introduction: Basics, applications, intranet and cloud, examples: Amazon, Google, Microsoft, IBM– advantages and disadvantages of cloud computing, Google app engine, Microsoft Azure, Amazon(EC2, S3,SQS),open stack, cloud computing services

Unit-II

Hardware and architecture: clients-security-network-services. Accessing the cloud: platforms-web applications-web APIs- web browsers. Cloud storage: overview-providers. Standards: application-client-infrastructure-service.

Unit-III

Software as Service: overview-driving forces-company offerings-industries. Software plus services: Overview-mobile device integration-providers-Microsoft Online.

Unit-IV

Developing Applications: Google-Microsoft-Intuit Quick Base-Cast Iron Cloud-Bungee Connect-Development (App engine, Azure, open stack etc.)- trouble shooting and application management.

Unit-V

Local clouds and thin clients: Virtualization-server solutions-thin clients. Cloud Migration: cloud services for individuals-enterprise cloud- methods for migration-analyzing cloud services.

Text Book:

1. Anthony T.Velte, Toby Velte ,"Cloud Computing a practical approach" , Mcgraw Hill, 2010.

References:

1. Janakiram M.S.V, "Demystifying the Cloud – An introduction to Cloud Computing",   version 1.1, 2010.

2. Mark C. Chu-Carroll, "Code in the Cloud- Programming Google App Engine", The Pragmatic

   Bookshelf Raleigh, North Carolina Dallas, Texas, 2011.

Course outcomes:

CO1: Acquire Knowledge on the features and development of Cloud Computing.
CO2: Define the principles of virtualization.
CO3: Use various performance criteria to evaluate the quality of the cloud architecture.
CO4: Identify the Service-Oriented Architecture for Distributed Computing workflow.
CO5: Create combinatorial auctions for cloud resources and design scheduling algorithms for computing clouds

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔		✔	
CO2			✔		✔				✔		✔	✔	✔
CO3	✔			✔			✔	✔		✔			✔
CO4	✔	✔			✔				✔			✔	
CO5			✔			✔	✔	✔			✔		

19MCAP506 - PROGRAMMING LAB-IX (Data Analytics: Python and R)

Learning objectives:

- To understand and be able to use the basic programming principles such as data types, variable, conditionals, loops, array, recursion and function calls.
- To learn how to use basic mathematical problems are evaluated and be able to manipulate text files and file operations.
- To understand the process and will acquire skills necessary to effectively attempt a programming problem and implement it with a specific programming language - Python.

List of Exercises

1. Python Program to check if a Number is Positive, Negative or Zero.
2. Python program to check prime numbers.
3. Python Program to check Armstrong Number.
4. Python Program to Find Hash of File.
5. Python Program to Root search.
6. R Program to Check if a Number is Odd or Even.
7. R Program to Find the Factors of a Number.
8. R Program to Convert Decimal into Binary using Recursion.
9. R Program to find Fibonacci Sequence Using Recursion.
10. R program to find the Factorial of a Number Using Recursion.

Course Outcomes:

CO1: Describe the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python.

CO2: Express different Decision Making statements and Functions in R and Python.

CO3: Interpret Object oriented programming in Python.

CO4: Understand and summarize different File handling operations in R.

CO5: Design and develop Client Server network applications using Python and R.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔	✔		
CO2			✔	✔	✔				✔			✔	✔
CO3	✔	✔					✔				✔		✔
CO4	✔				✔		✔		✔	✔		✔	
CO5			✔			✔		✔	✔		✔		✔

DEPARTMENT ELECTIVES

19MCAE406: OBJECT ORIENTED ANALYSIS AND DESIGN

Learning objectives:

- To develop deep understanding of OOPs language features.
- To construct strong concepts of OOP's and implement the same using UML diagrams.
- To specify, analyze and design the use case driven requirements for a particular system and Model the event driven state of object and transform them into implementation specific layouts.
- To identify, analyze the subsystems, various components and collaborate them interchangeably.
- To develop the skill among the students to analyze and design a complete system through case studies.

Unit-I

Complexity: Introduction- Object Basics-OOA-OOD-OO Modeling – Object Oriented Systems development life cycle-The Inherent Complexity of Software-The Structure of Complex Systems- On Designing Complex Systems.

Unit-II

Classes and Objects: The Nature of an Object- Relationships among Objects-The Nature of a Class- Relationships among Classes-The Inter play of classes and objects-On building quality classes and objects.

Unit-III

Classification: The Importance of Proper Classification-Identifying Classes and Objects- Key Abstractions and Mechanisms.

The Notation: Elements of the Notation-Class Diagrams-State Transition Diagrams- Object Diagrams-Interaction Diagrams-Module Diagrams-Process Diagrams-Activity Diagram-ComponentDiagram-DeploymentDiagram-UseCaseDiagram-Appling the Notation.

Unit-IV

The Process: First Principle-The Micro Development Process-The Macro Development Process.

Pragmatics: Management and Planning-Staffing-Release Management-Reuse-Quality Assurance and Metrics-Documentation-Tools-Special Topics-The Benefits and Risks of Object Oriented Development.

Unit-V : Analysis-Design-Evolution and Maintenance of:

1) Data Acquisition: Weather Monitoring Station.

2) Frameworks: Foundation Class library and

3) Client/Server Computing: Inventory Tracking.

Text Book:

1.GradyBooch,"ObjectOrientedAnalysisand Design with Applications",the Benjamin

CummingsPublishingCompanyInc.,SecondEdition,1994.

References:

1. Taylor.D.,"Object Oriented Information Systems",John Wiley and Sons,1992.

2.Pinson.L.and Wiener R.," Application of Object Oriented Programming", Addison WesleyPublishingCompany,1990.

3. AliBahrami, "Object Oriented Systems Development", Irwin Mcgraw Hill, InternationalEdition,1999.

Course outcomes:

CO1: Recognize, diagram, and implement introductory concepts of Object Oriented Programming.
CO2: Analyze, design and document the requirements of the system.
CO3: Identify, analyze, and model structural and behavioral concepts of the system.
CO4: Develop and explore the conceptual model into various scenarios and applications.
CO5: Apply the concepts of architectural design for deploying the code for software.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔		✔		✔			✔
CO2			✔		✔			✔	✔		✔		✔
CO3		✔					✔					✔	
CO4	✔		✔		✔				✔			✔	
CO5		✔				✔	✔	✔		✔	✔		

19MCAE407: OPEN SOURCE TECHNOLOGIES

Learning objectives:

- To learn the concepts of dynamic web content.
- To learn implementation of web development server.
- To understand the basics of syntax, arrays, functions and objects in PHP programming language.
- To know the structure of MySQL database and access using PHP.
- Testing and debug a PHP applications.

UNIT-I

Introduction to Dynamic Web Content: HTTP and HTML: Berners-Lee's Basics - The Request/Response Procedure - The Benefits of PHP, MySQL, JavaScript, CSS, and HTML5 - Using PHP - Using MySQL - Using JavaScript - Using CSS - And Then There's HTML5 - The Apache Web Server - About Open Source. Setting Up a Development Server: WAMP, MAMP, or LAMP - Installing XAMPP on Windows - Testing the Installation - Installing XAMPP on Mac OS X - Accessing the Document Root - Installing a - LAMP on Linux - Working Remotely - Logging In - Using FTP - Using a Program Editor - Using an IDE.

UNIT-II

Introduction to PHP: - Incorporating PHP Within HTML - The Structure of PHP - Using Comments -Basic Syntax - Variables – Operators - Variable Assignment - Multiple-Line Commands -Variable Typing - Constants - Predefined Constants - The Difference Between the echo and print Commands - Functions - Variable Scope. Expressions and Control Flow in PHP: Expressions - Operators – Conditionals – Looping - Implicit and Explicit Casting - PHP Dynamic Linking - Dynamic Linking in Action

UNIT-III

PHP Functions and Objects: PHP Functions - Including and Requiring Files - PHP Version Compatibility - PHP Objects. PHP Arrays: Basic Access - The foreach...as Loop - Multidimensional Arrays - Using Array Functions.

UNIT-IV

Practical PHP: Using printf - Date and Time Functions - File Handling - System Calls - XHTML - HTML5 Introduction to MySQL: MySQL Basics - Summary of Database Terms - Accessing MySQL via the Command Line – Indexes - MySQL Functions - Accessing MySQL via phpMyAdmin.

UNIT-V

Accessing MySQL Using PHP: Querying a MySQL Database with PHP - Example - Practical MySQL - Preventing Hacking Attempts - Using mysqli procedurally Form Handling: Building Forms - Retrieving Submitted Data. Cookies, Sessions, and Authentication: Cookies in PHP - HTTP Authentication - Sessions.

Text book

1. Robin Nixon, "Learning PHP, MySQL & JavaScript With jQuery, CSS & HTML5", Forth edition, Published by O'Reilly Media, December 2014.

References:

1. Elizabeth Naramore, Jason Gerner, Yann Le Scouarnec, Jeremy Stolz, Michael K. Glass, "Beginning PHP5, Apache, and MySQL Web Development", Published by Wiley Publishing, Inc., 2005.
2. Tim Converse and Joyce Park with Clark Morgan, "PHP5 and MySQL Bible", Published by Wiley Publishing, Inc., 2004.

Course outcomes:

CO1: Apply dynamic web content concept into real time web applications.
CO2: Develop web server side programming.
CO3: Design database for real time applications.
CO4: How to receive and process form submission data.
CO5: Using PHP built-in functions and creating custom functions

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔	✔	✔		✔	
CO2	✔		✔		✔		✔			✔		✔	✔
CO3		✔					✔		✔				✔
CO4		✔			✔						✔	✔	
CO5	✔	✔	✔			✔		✔		✔			✔

19MCAE408: SOFT COMPUTING

Learning objectives:

- To introduce the techniques of soft computing.
- To explain the hybridization of soft computing systems which differ from conventional AI.
- Computing in terms of its tolerance to imprecision and uncertainty.
- Develop the skills to gain a basic understanding of neural network theory and fuzzy logic theory.
- Introduce students to artificial neural networks and fuzzy theory from an engineering perspective.

Unit-I

Introduction - Neural Networks Characteristics – History of development in neural networks-artificial neural net terminology –Model of a neuron -Types of learning. Supervised-Unsupervised learning- Perceptrons -Architecture of a Perceptron – Perceptron convergence algorithm- Generalized delta rule for weight adjustment - Theory of Back propagation Training Algorithm -Rate of Learning- Training Considerations-Characteristics of BP Learning Algorithm- Limitations of BP Learning -Accelerated convergence of BP through learning-rate adaptation.

Unit-II

Learning-UnsupervisedLearning-HebbianLearning-CompetitiveLearning – Boltzmann Learning-Supervised Learning-Error-Correction learning-Reinforcement Learning- Recurrent Network-Basic Concepts-Hopfield Network-Operation Features of Hopfield Network-Error Performance of Hopfield Network –Storage Capacity of Hopfield Network.

Unit-III

Radial basis function neural networks – Basic learning laws in Radial basis function nets-Counter propagation networks – Adaptive resonance theory networks – Applications of neural nets such as pattern recognition –Optimization-Associative memories-speech and decision-making.

Unit-IV

Fuzzy Logic- Basic concepts of Fuzzy Logic– Fuzzy set versus Crisp Set-Linguistic variables-membership functions- operations of fuzzy sets – Fuzzy IF-THEN rules- fuzzy relations-fuzzy conditional statements - fuzzy rules- fuzzy learning algorithms- applications of fuzzy logic.

Unit-V

Neuro – fuzzy and fuzzy - neural control systems – Adaptive fuzzy systems – optimizing the membership functions and the rule base of fuzzy logic controllers using neural networks- fuzzy transfer functions in neural networks.

Text Books:

1.Haykin. S,"NeuralNetworks:AComprehensiveFoundation", 2nd Ed, Prentice Hall. 1999.

2. TimothyJ.Ross, "FuzzyLogic EngineeringApplications", McGrawHill, NewYork,1997.

References:

1.WassermanP.D,"Neural Computing Theory and Practice," VanNo strand   Reinhold, NewYork,1997.

2.RizaCBerkinandTrubatch, ʹFuzzysystemsDesignPrinciplesʺ,BuildingaFuzzyIF,THENRuleBases,IEE EPressISBN0-7803-1151-.5.

3. Kosko,B,"NeuralNetworksandFuzzySystems:ADynamicalApproachto Machine Intelligence", PrenticeHall,NewDelhi,1991.

Course outcomes:

CO1: Comprehend the fuzzy logic and the concept of fuzziness involved in various systems and fuzzy set theory.

CO2: Understand the concepts of fuzzy sets, knowledge representation using fuzzy rules, approximate reasoning, fuzzy inference systems, and fuzzy logic.
CO3: To understand the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations.
CO4: Understand appropriate learning rules for each of the architectures and learn several neural network paradigms and its applications.
CO5: Reveal different applications of these models to solve engineering and other problems.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔		✔	✔		✔		✔					✔
CO2			✔		✔				✔		✔	✔	✔
CO3			✔				✔					✔	✔
CO4	✔	✔		✔	✔				✔	✔	✔	✔	
CO5		✔	✔			✔		✔		✔	✔		

19MCAE409- DATA MINING TECHNIQUES

Learning Objectives:

- To introduce concepts of data mining techniques.
- To understand its applications in knowledge extraction from databases.
- To develop skills of using recent data mining software for solving practical problems.
- To understand preprocessing operations on data.
- To understand data interpretation, transformation and reduction techniques.

Unit-I

Data mining – Motivation – Importance - DM Vs KDD - DM Architecture - Data Types – DM Tasks –DM System Classification - Primitives of DM - Data Mining Query Language - DM Metrics - DM Applications - DM Issues – Social Implications of DM.

Unit-II

Data Preprocessing: Summarization - Data cleaning - Data Integration and Transformation - Data Reduction - Discretization and Concept Hierarchy Generation.

Unit-III

Mining Frequent Patterns – Frequent Item set Mining Methods. Classification: Classification by Decision Tree Induction – Bayesian Classification – Rule based Classification - Prediction– Accuracy and Error Measures.

Unit-IV

Cluster Analysis – Types of Data in Cluster Analysis – Categorization of clustering Methods – Partition Methods - Outlier Analysis – Mining Data Streams – Social Network Analysis – Mining the World Wide Web.

Unit-V

Data Warehousing: OLTP Vs OLAP - Multidimensional Data Model -DW Architecture Efficient Processing of OLAP queries - Metadata repository – DWH Implementation – OLAM.

Text book:

1. JiaweiHan, Micheline amber, "Data Mining: Concepts and Techniques", 3rd Edition, Elsevier India Private Limited, 2012

References:

1. Margaret H. Dunham, "Data Mining: Introductory and Advanced Topics", Pearson Education, 2012.

2. K.P.Soman, ShyamDiwakar, V.Ajay, "Insight into Data Mining Theory & Practice, Prentice Hall India, 2012

3. G.H.Gupta, "Introduction to Data Mining with Case Studies", 2nd Edition, PHI.

4. Ralph Kimball, Margy Ross "The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling", 3rd Edition ,wiley , Jul 2013

Course Outcomes:

CO1: Explain the concepts in data mining and KDD, recognizing issues in Data Mining.

CO2: Practice the preprocessing operations of Data.

CO3: Define the methodologies in Data interpretation, transformation and reduction.

CO4: Perform Association Rule Mining, Classify and Cluster the data sets into groups.

CO5: Implement star schema through ETL tools.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔				✔	✔			✔	
CO2			✔		✔		✔			✔			✔
CO3				✔		✔			✔		✔		✔
CO4	✔	✔			✔		✔			✔	✔	✔	
CO5		✔	✔			✔		✔					✔

19MCAE506: NETWORKSECURITY

Learning objectives:

- To understand the fundamentals of Cryptography.
- To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
- To understand the various key distribution and management schemes.
- To understand how to deploy encryption techniques to secure data in transit across data Networks.
- To design security applications in the field of Information technology.

Unit-I

Symmetric Ciphers: Classical Encryption Techniques – Block Ciphers and the Data Encryption Standard – Finite Field – Advanced Encryption Standard– Symmetric Ciphers–Confidentiality using Symmetric Encryption.

Unit-II

Public Key Encryption and Hash Functions: Introduction to Number Theory– Public Key Cryptography and RSA – Key Management – other Public Key Cryptosystem– Message Authentication and Hash Functions – Hash and MAC Algorithms–Digital Signatures and Authentication Protocols.

Unit-III

Program Security: Secure Programs – Non Malicious Program Errors – Viruses and Others Malicious Code–Targeted Malicious Code–Control Against Threats.

Unit-IV

Database Security: Introduction to Database – Security Requirement – Reliability and Integrity – Sensitive Data–Inference–Multilevel Databases- Multilevel Security

Unit-V

Network Security: Networks concepts – Threats in Networks – Network Security Controls – Firewalls –Electronic Mail Security– IP Security – Web Security.

Text Books:

1. Charles B. Pfleeger -Shari Lawrence Pfleeger ,"Security in Computing", Third Edition, PearsonEducation,2003.
2. William Stallings, "Cryptography and Network Security – Principles and Practices", Pearson Education, Fourth Edition, 2003.

Course outcomes:

CO1: Analyze the vulnerabilities in any computing system.
CO2: Able to design a security solution.

CO3: Identify the security issues in the network and resolve it.

CO4: Evaluate security mechanisms using rigorous approaches, including theoretical.

CO5: Compare and Contrast different IEEE standards and electronic mail security.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔	✔	✔			✔		✔
CO2	✔		✔		✔				✔		✔		✔
CO3	✔		✔		✔		✔			✔			✔
CO4		✔		✔	✔				✔			✔	
CO5	✔	✔	✔			✔		✔		✔		✔	

19MCAE507: INTERNET OF THINGS

Learning objectives:

- To understand the fundamentals of Internet of Things
- To learn about the basics of IOT protocols
- To build a small low cost embedded system using Raspberry Pi.
- To apply the concept of Internet of Things in the real world scenario.
- To classify Real World IoT Design Constraints, Industrial Automation in IoT.

Unit I

Introduction to IoT: Introduction - Definition & Characteristics of IoT - Physical Design of IoT - Things in IoT - IoT Protocols - Logical Design of IoT - IoT Functional blocks - IoT Communication Models - IoT Communication APIs - IoT Enabling Technologies - Wireless Sensor Networks - Cloud Computing - Big Data Analysis - Communication protocols - Embedded Systems - IoT Levels & Deployment Templates. IoT and M2M: Introduction - M2M - Difference between IoT and M2M-SDN and NFV for IoT. Ubiquitous IoT Applications**:** A Panoramic View of IoT Applications-Important Vertical IoT.

Unit II

Four pillars of IoT: The Horizontal, Verticals, and Four Pillars, M2M, RFID, WSN, SCADA. The DNA of IoT - Device, Connect and Manage - Device: Things That Talk - Connect: Via Pervasive Networks - Manage: To Create New Business Value.

Unit III

Middleware and IoT: An Overview of Middleware - Communication Middleware for IoT. Protocol standardization of IoT **:** Web of Things versus Internet of Things - IoT Protocol Standardization Efforts- Unified Data Standards: A Challenging Task.

Unit 1V
Architecture Standardization for WoT: Platform Middleware for WoT - Unified Multitier WoT Architecture - WoT Portals and Business Intelligence - Challenges of IoT Information Security. The Cloud of Things **:** Cloud Middleware - NIST's SPI Architecture and Cloud Standards - Cloud Providers and System. The Cloud of Things:The Internet of Things and Cloud Computing, Mobile Cloud Computing. MAI versus XaaS**:** The Long Tail and the Big Switch - The Cloud of Things Architecture.

Unit V

Thinking about Prototyping: Prototypes and Production - Open Source versus Closed Source. Prototyping Embedded devices **:** Electronics - Embedded Computing Basics - Arduino, Raspberry pi - Beagle Bone Black - Electric Imp - Other Notable Platforms.

Text Books:

1. Arshdeep Bahga, Vijay Madisetti, —Internet of Things: A Hands of Approach‖ , Arshdeep Bagha & Vijay Madisetti, First Edition, 2014.

2. Honba Zhou, —The Internet of things in the Cloud: A Middleware Perspective‖, CRC Press, First Edition, 2012.

3**.** Adrian McEwen, Hakim Cassimally, — Designing The Internet things‖, John Wiley and Sons, First Edition, 2014.

Course outcomes:

CO1: Interpret the vision of IoT from a global context.

CO2: Determine the Market perspective of IoT.

CO3: Compare and Contrast the use of Devices, Gateways and Data Management in IoT.

CO4: Implement state of the art architecture in IoT.

CO5: Illustrate the application of IoT in Industrial Automation and identify Real World Design Constraints.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔			✔		✔		✔		✔		✔	
CO2			✔		✔				✔		✔		✔
CO3	✔			✔			✔			✔			
CO4	✔				✔			✔	✔	✔	✔		
CO5			✔			✔	✔					✔	✔

19MCAE508: DIGITAL IMAGE PROCESSING

Learning objectives:

- To understand the fundamental concepts related to image processing, feature extraction, pattern analysis etc.

- To understand image processing, analysis and understanding.

- To understand low-level, middle-level and high-level operations.

- To apply the concepts to solve computer vision problems of different fields.

- To study the image compression procedures.

Unit-I

Digital Image Processing Systems: Introduction-Structure of human eye-Image formation in the human eye-Brightness adaptation and discrimination-Image sensing and acquisition-Storage-Processing-Communication-Display. Image sampling and quantization- Basic relationships between pixels

Unit-II

Image Enhancement in the Spatial Domain: Gray level transformations-Histogram processing-Arithmetic and logic operations-Spatial filtering: Introduction - Smoothing and sharpening filters

Image Enhancement in the Frequency Domain: Frequency domain filters: Smoothing and Sharpening filters-Homomorphic filtering

Unit-III

Wavelets and Multi resolution Processing: Image pyramids-Sub and coding-Haar transform-Series expansion-Scaling functions- Wavelet functions-Discrete wavelet transforms in one dimensions- Fast wavelet transform-Wavelet transforms in two dimensions

Unit-IV

Image Data Compression: Fundamentals-   Redundancies: Coding- Inter pixel- Psycho-visual-Fidelity criteria-Image compression models-Error free compression- Lossy compression-Image compression standards: Binary image and Continuous tone still image compression standards-Video compression standards.

Unit-V

Morphological Image Processing: Introduction-Dilation- Erosion-   Opening- Closing-Hit-or-Miss transformation-Morphological algorithm operations on binary images-Morphological algorithm operations on gray-scale images.

Image Segmentation: Detection of discontinuities-Edge linking and Boundary detection-Thresholding-Region based segmentation

Image Representation and   Description: Representation schemes-Boundary descriptors-Regional descriptors

Text Books:

1. Gonzalez R.C and R.E.Woods**, "**Digital Image Processing", Second Edition, Pearson Education2002

2. AnilK. Jain, "Fundamentals of Image Processing",PHINewDelhi2001.

References:

1. William Pratt, "Digital Image Processing"**,** John Wiley & Sons, Third edition, 2001.

2. Hany Farid, "Fundamentals of Image Processing"**,** Dartmouth College, 2010.

3. Stefan G. Stanciu, "Digital Image Processing" ,In Tech **,** 2012.

4. Dwayne Philipps," Image processing in C: Analysis and Enhancing Digital Images",R & D Books**,** 1997.

Course outcomes:

CO1: Review the fundamental concepts of a digital image processing system.
CO2: Analyze images in the frequency domain using various transforms.
CO3: Evaluate the techniques for image enhancement and image restoration.
CO4: Categorize various compression techniques.
CO5: Interpret image segmentation and representation techniques.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔		✔			✔			✔	
CO2	✔		✔		✔			✔			✔		✔
CO3				✔			✔		✔	✔			✔
CO4	✔	✔			✔		✔					✔	
CO5		✔	✔			✔		✔		✔	✔		

19MCAE509: PATTERN RECOGNITION

Learning Objectives:

- Understand the concept of a pattern and the basic approach to the development of pattern recognition and machine intelligence algorithms.

- Understand the basic methods of feature extraction, feature evaluation, and data mining.

- Understand and apply both supervised and unsupervised classification methods to detect and characterize patterns in real-world data.

- Develop prototype pattern recognition algorithms.

- It can be used to study algorithm behaviour and performance against real-world multivariate data.

Unit-I

Introduction: Machine perception -pattern recognition systems-design cycle–learning and adaptation. Bayesian decision theory: Continuous features– minimum-error-rate classification-classifiers,discriminantfunctions,anddecisionsurfaces-normal density

-discrete features-Bayesian belief networks.

Unit-II

Maximum-likelihood and   Bayesian parameter estimation: Maximum-likelihood estimation-Bayesian estimation –Bayesian parameter estimation: Gaussian case-Problems of dimensionality.

Unit-III

Component analysis   and discriminants: Principal component analysis- fisher linear

discriminant-multiplediscriminantanalysis.Expectation-maximization algorithm. Hidden Markov models: Evaluation–decoding-learning.

Unit-IV

Nonparametric techniques: Probabilistic neural networks - k-nearest-neighbor rule. Linear discriminant functions and decision surfaces. Support vector machines: Training- multi category generalizations.

Unit-V

Neural networks: Biological and artificial neurons- perceptron training algorithm- back propagation training algorithm and error surfaces- radial basis function neural network training algorithm-Decision trees-classification and regression trees- K-means clustering-fuzzyk-means clustering. Neural network approach to component analysis: Principal component analysis – nonlinear component analysis-independent component analysis.

Text Books:

1. Duda R.O,E.Hart,andD.G.Stork, "Patternclassification",secondedition,John Wiley&Sons, Singapore,2003.

2. Haykin S, Neural networks: A comprehensive foundation, second edition, Pearsoneducation,Singapore,2001.

References:

1. Vapnik V, "Statistical learningtheory", JohnWiley&Sons, NewYork,1998.

2. Burges Christopher J.C, "A tutorial on support vector machines for pattern recognition", Data mining and knowledgediscovery,pp.121-167,1998.

Course outcomes:

CO1: Fundamentals of Pattern Recognition Various Statistical, Syntactic and applications.

CO2: Neural Network Approaches of PR and their applications.

CO3: Understand the fundamentals of Pattern Recognition. Learn the various approaches to identify the patterns.

CO4: Applying the appropriate techniques on the real time application development.

CO5: Understand the principles of Bayesian parameter estimation and apply them in relatively simple probabilistic models.

Outcome Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	✔	✔		✔				✔		✔	✔		
CO2			✔		✔		✔					✔	✔
CO3		✔		✔		✔		✔			✔		✔
CO4	✔	✔			✔	✔			✔			✔	
CO5		✔	✔				✔		✔	✔			

INTER DEPARTMENT ELECTIVE COURSE OFFERED TO OTHER DEPARTMENTS IN THIRD SEMESTER

19SCISX01 R PROGRAMMING

LEARNING OBJECTIVES

LO1. To provide an overview of a new language R used for data science and to introduce students to the R programming environment and related eco-system and thus provide them with an in demand skill-set, in both the research and business environments.

LO2. To demonstrate usage of as standard Programming Language.

LO3. To familiarize students with how various statistics like mean median etc. can be collected for data exploration in R and enable students to use R

UNIT-I:

Overview and Preliminaries

Basic Features of R - Design of R System-Limitations of R – Installation – R studio - Getting started with R interface - Entering Input - R Objects – Attributes - Creating Vectors - Mixing Objects – Matrices – Lists – Factors - Data Frames.

UNIT-II:

Input, Output, Reading and Subsetting

Reading Data Files - Reading in Larger Datasets - Calculating Memory Requirements - File Connections - Reading Lines of a Text File - Reading From a URL Connection - Subsetting a Vector - Subsetting a Matrix - Subsetting Lists - Subsetting Nested Elements of a List - Extracting Multiple Elements of a List - Partial Matching. Date, Time and Managing Data Frames Operations on Dates and Times - Data Frames - dplyr Package - Installing the dplyr package – select – filter – arrange – rename – mutate - group_by - pipeline operater.

UNIT-III:

Control Structures and Loop Functions

if-else - for Loops - Nested for loops - while Loops - repeat Loops - next, break - Looping on the Command Line - lapply() - sapply() - split() - Splitting a Data Frame – tapply - apply() - Col/Row Sums and Means - mapply() - Vectorizing a Function.

UNIT-IV:

Statistics functions - Debugging, Profiling

Mean - Median - variance of the population - Estimated standard deviation - Standard scores – Sort – Rank – summary function - Debugging Tools in R - traceback() - debug() - recover() - Using system.time() - Timing Longer Expressions - The R Profiler - Using summaryRprof().

UNIT-V:

Simulation and Graphs

Generating Random Numbers - Setting random number seed - Simulating Linear Model - Loading and Processing Raw Data – Creating a Graph - density plots - dot plots, bar charts - line charts - pie charts - box plots - Scatter plots.

TEXT BOOK

1. Roger D. Peng, "R Programming for Data Science", Lean Publishing, (2015), ISBN: 9781365056826, 1365056821.

REFERENCE BOOKS

1. Winston Chang, "R Graphics Cookbook", O'Reilly Media, Inc., (2012), ISBN: 9781449363086.
2. Using R for Introductory Statistics by John Verzani, CRC Press, 2004.

COURSE OUTCOMES

CO1. Install and use R for simple programming tasks.
CO2. Extend the functionality of R by using add-on packages and extract data from files and other sources and perform various data manipulation tasks on them.
CO3. Code statistical functions in R and use R Graphics and Tables to visualize results of various statistical operations on data.
CO4. Apply the knowledge of R gained to data Analytics for real life applications. to conduct analytics on large real life datasets.

OUTCOME MAPPING

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	3		3		3		3		3	3		
CO2			3		3		3					2	2
CO3				3				2			3		1
CO4	3	2			3				2		1	1	

VALUE ADDED COURSES OFFERED BY OUR DEPARTMENT

SCISVAC01: Web Development

Learning Objectives

LO1. To impart basic knowledge on web development

LO2. To impart basic scripting ideas and importance in client server architecture

LO3. To provide the basic knowledge about ASP, XML, CSS, PHP and MySql.

Unit-1

HTML: History of the Internet and World Wide Web – HTTP, SMTP, POP3, MIME, Understanding roles of Web Browsers and Web Servers. Structure of HTML, Text formatting, Text styles, hyper link, image, and tables.

Unit-2

Frames, Forms and CSS: Frames, Forms and controls, Embedding audio, video and animated files in HTML, CSS –Understanding CSS, Internal CSS, External CSS, Font Properties, Text Properties, Color and Background properties, Table properties, Numbering and List Properties.

Unit-3

JavaScript: Data types and literals, operators, conditional statements, loop constructs, reserved words; core Objects: Array Object, Date Object; Functions: passing value to JavaScript functions, user defined functions, Handling old browsers , java script events, formatting cookie, retrieving cookie value from the cookie file, removing a cookie , animations using events.

Unit-4

ASP & XML: ASP – Working of ASP – Request and Response Objects – File System Objects – Session tracking and cookies – ADO – Access a Database from ASP – XML – Structure in Data – Name spaces – DTD – Vocabularies – Accessing Web servers – IIS – Apache web server.

Unit-5

▪ **PHP & MySQL:** Why PHP and MySQL - Server-Side Web Scripting - Getting Started with PHP - Adding PHP to HTML -Syntax and Variables - Control and Functions - Passing Information between Pages – Strings – Arrays and Array Functions – Numbers - MySQL Database Administration - PHP/MySQL Functions - Displaying Queries in Tables - Building Forms from Queries.

Text Books

1. Kris Jamsa, konrad King and Andy Anderson, "HTML & Web Design Tips and Techniques", Tata McGraw-Hill, First Edition, 2002.

2. Powell T.A, HTML: Complete Reference, Tata McGraw-Hill, Fifth Edition, 2010.

Reference Books

1. Deitel & Deitel, Goldberg, Internet and World Wide Web – How to Program, Third Edition, Pearson Education Asia, 2005.

2. Elizabeth Naramore, Jason Gerner, Yann Le Scouarnec, Jeremy Stolz, and Michael K. Glass, "Beginning PHP5, Apache, and MySQL Web Development", First Edition, Wrox publications, 2005.

3. Rajkamal, "Web Technology", First Edition, Tata McGraw-Hill, 2001. Tim Converse, Joyce Park and Clark Morgan "PHP5 and MySQL Bible", Wiley Publishing, Inc. 2004.

Course Outcomes

On successful completion of the course, the students will be able to
CO1. Develop simple components in web pages.
CO2. Write code using scripting languages.
CO3. Can connect databases via web applications.
CO4. Design a web page.

SCISVAC02: Fundamentals of Computing

Learning Objectives

LO1. To impart basic knowledge on Microsoft office

LO2. To provide the basic knowledge about Graphs and Charts.

Unit–1: Document Creation in MS-WORD
Creating new Documents –Document template – Typing Text – Editing Text – Insert Text – Go to, Find, and Replace text or word- Formatting of Text – Format painter – Cut and Paste utility.

Unit–2: Table Creation in Ms-Word
Creating tables – Working with table – Merging cells – Splitting a cell – Splitting a table -using auto correct.

Unit–3: Ms-PowerPoint
Create Slide Presentation – Design Theme – Add Text – Editing Techniques – Slide Master – Format Slide.-Transition and Animation.

Unit–4: Ms-Excel

The typical worksheet or spread sheet – cell and their properties – formatting cell – text, numbers, currency, accounting, date, time, percentage, scientific – formats.

Formula: using arithmetic and relational operators in a worksheet -Advanced Formulas: sum, count, Average, Max, Min, Product.

Unit-5: Graphs and Charts

Bar diagrams, pie charts, Area, - Building Line Diagrams, Histograms, Scatter plots - Frequency Graphs.Introduction to Flash.

Text Book:

    1. Torben Lage Frandsen ,'Microsoft Office Word 2007', Bookboon, India.

Reference Book:

    1. Stephen Moffat ,'Microsoft Word 2010', Bookboon, India.

Course Outcomes

On successful completion of the course, the students will be able to

CO1.   Develop simple Mail Merge.

CO2.   Design presentation.

CO3.   Spreadsheet Data Collection.

CO4.   Convert data into charts or graphs

SCISVAC03: Advanced Web Development

Learning Objectives

LO1.To impart advanced ideas of web development

LO2.To impart scripting ideas and importance in web development.

LO3.To provide the basic knowledge about ASP.NET, ADO.NET and web services.

Unit-1

Overview of ASP.NET - The .NET framework – Learning the .NET languages : Data types – Declaring variables- Scope and Accessibility- Variable operations- Object Based manipulation- Conditional Structures- Loop Structures- Functions and Subroutines. Types, Objects and Namespaces : The Basics about Classes- Value types and Reference types- Advanced class programming- Understanding name spaces and assemblies. Setting Up ASP.NET and IIS   .

Unit–2

Developing ASP.NET Applications - ASP.NET Applications: ASP.NET applications– Code behind- The Global.asax application file- Understanding ASP.NET Classes- ASP.NET Configuration. Web Form fundamentals: A simple page applet- Improving the currency converter- HTML control classes- The page class- Accessing HTML server controls. Web

controls: Web Control Classes – Auto PostBack and Web Control events- Accessing web controls. Using Visual Studio.NET: Starting a Visual Studio.NET Project- Web form Designer- Writing code- Visual studio.NET debugging. Validation and Rich Controls: Validation- A simple Validation example- Understanding regular expressions- A validated customer form. State management - Tracing, Logging, and Error Handling.

Unit–3

Working with Data - Overview of ADO.NET - ADO.NET and data management- Characteristics of ADO.NET-ADO.NET object model. ADO.NET data access : SQL basics– Select , Update, Insert, Delete statements- Accessing data- Creating a connection- Using a command with a DataReader - Accessing Disconnected data - Selecting multiple tables – Updating Disconnected data. Data binding: Single value Data Binding- Repeated value data binding- Data binding with data bases. Data list – Data grid – Repeater – Files, Streams and Email – Using XML

Unit–4

Web Services - Web services Architecture : Internet programming then and now- WSDL– SOAP- Communicating with a web service-Web service discovery and UDDI. Creating Web services : Web service basics- The StockQuote web service – Documenting the web service- Testing the web service- Web service Data types- ASP.NET intrinsic objects. Using web services: Consuming a web service- Using the proxy class- An example with TerraService.

Unit–5

Advanced ASP.NET - Component Based Programming: Creating a simple component – Properties and state- Database components- Using COM components. Custom controls: User Controls- Deriving Custom controls. Caching and Performance Tuning: Designing and scalability– Profiling- Catching- Output catching- Data catching. Implementing security: Determining security requirements- The ASP.NET security model- Forms authentication- Windows authentication.

Text Book

1. Mathew Mac Donald, "ASP.NET Complete Reference", TMH 2005.

Reference Books

1. Crouch Matt J, "ASP.NET and VB.NET Web Programming", Addison Wesley 2002.

2. J.Liberty, D.Hurwitz, "Programming ASP.NET", Third Edition, O'REILLY, 2006.

Course Outcomes

On successful completion of the course, the students will be able to

CO1. Overall view to develop web pages.

CO2. Write advanced codes in scripting languages.

CO3. Connect the different databases via web applications.

CO4. Able to use components, remote calls and web services in distributed web applications.

SCISVAC04: Internet and its applications

Learning Objectives

LO1.   To impart ideas of online communication

LO2.   To provide the basic knowledge about Social Medias

Unit I : Internet concepts

Introduction – Internet Connection Concepts – Connecting to Dial-up Internet Accounts – High Speed Connections : ISDN, ADSL, and Cable Modes – Intranets : Connecting LAN to the Internet.

Unit II: E-mail concept

E-mail Concepts – E-mail Addressing – E-mail Basic Commands – Sending and Receiving Files by e-mail – Controlling e-mail Volume – Sending and Receiving secure e-mail.

Unit III: Internet services

Conferencing Concepts – E-mail Mailing Lists – Usenet Newsgroup Concepts – Reading Usenet Newsgroups – Video Conferencing.

Unit IV : Web concepts and Browsers

World-Wide-Web Concepts – Elements of Web – Clients and Servers – URL and –Web Browsers – Netscape Navigator and Communicator-Microsoft Internet Explorer.

Unit V: Social Media

Online Chatting – Whatsapp –Telegram- Instagram – Facebook – Twitter.

Text Book:
1. Alexis Leon and Mathews Leon- 2012Internet for everyone, Leon Techworld, 2000 Chennai, India.

Reference Book:
1. Kamlesh N. Agarwal & PrateekA.Agarwal –An introduction to Wireless application protocol, McMillan India Ltd

Course Outcomes

On successful completion of the course, the students will be able to

CO1.   Overall view of mailing concepts.

CO2.   Communicate via social media.