BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY), PUNE

Faculty of Science
M.Sc - Computer Science
New Syllabus

"Social Transformation Through Dinamic Education"

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)

YASHWANTRAO MOHITE COLLEGE OF ARTS, SCIENCE AND COMMERCE,

PUNE 411038

Accredited with 'A+' Grade (2017) by NAAC
'A' Grade University Status by MHRD, Govt. of India
Accredited (2004) & Reaccredited (2011) with 'A' Grade by NAAC

MASTER OF COMPUTER SCIENCE (M.Sc. Computer Science) PROGRAMME

CBCS 2018 COURSE STRUCTURE

Under the Faculty of Science

TO BE IMPLEMENTED FROM ACADEMIC YEAR 2018-19

BHARATI VIDYAPEETH
DEEMED UNIVERSITY, PUNE (INDIA)

Master of Computer Science M.Sc. (Computer Science)
(CBCS 2018 COURSE)

Under: The Faculty of Science
(To be implemented from June 2018)

The Master of Computer Science, M.Sc. (Computer Science) Program is a full time 94 Credits program offered by Bharati Vidyapeeth Deemed University (BVDU), Pune. The expectations and requirements of the Software Industry are visualized while designing the M.Sc. (Computer Science).

1. Objectives: M.Sc. (Computer Science) Course:

- The main objective of the M.Sc. Computer Science course is to provide the students a clear understanding of the basic concepts and principles of Computer Science as a discipline and the rich and specialized skill sets required to handle the computing systems in an applied branch of knowledge that represent a realistic world.
- All aspects of knowledge acquisition, storage, processing, presentation and transition will be covered in the course.
- Provision for advanced knowledge in Computer Science, exposure to the practical and theoretical concepts of computing, current and emerging trends in technology in the context of networking environment are the major attractions of the programme.
- The course covers almost all the core subjects in Computer Science like Design and Analysis of Algorithm, Network Programming, OOP, Software Engineering, Database Concepts etc, in addition to a few elective subjects.
- The Case Studies in all the three semesters intensively focus on the systematic design and implementation of various technological applications through which the students attain the skills in system development and team spirit.
- As part of the course, the students are advised to take live projects of 120 to 140 days duration from established institutions for implementation during their project term.
- In addition, 'communication skills', 'life skills' which are necessary for career growth and for leading quality life are also imparted.

Learning Outcomes from the M.Sc. (Computer Science)

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

(a) Analyze problems and design effective and efficient software solutions
(b) Develop software under latest Application Development Environments.
(c) Learn new technologies with efficient implementation
(d) Read, write, and contribute to technical literature
(e) Work in team
(f) Socially committed IT professionals

2. Eligibility for Admission to the Course:

A student shall be eligible for admission to the First Year M.Sc. (Computer Science) degree course who has completed B.Sc. (Computer Science) / B.Sc. (Computer Application)/B.Sc. (IT) graduation from any recognized university satisfying the following conditions. The candidate should have secured at least 50% (45% for SC/ST) in aggregate at graduate level university examination.

3. Intake Capacity:

The intake capacity of the course will be 50 seats every year.

4. The broad objectives of the Learning Outcomes-based Curriculum Framework (LOCF) of Master of Science Degree programme in Computer Science

Master's Degree is the well-recognized postgraduate qualification in higher education. The contents of this degree are determined in terms of knowledge and understanding, expertise and skills that a student intends to acquire. Often it does not come within the traditional boundaries recognizable at previous academic levels of study; it is specialised and close to the boundaries of current knowledge.

Master's Degree programmes attract entrants with a bachelor's degree with honors or equivalent, or experience that may or may not be directly relevant to the particular profession. Thus, M.Sc.(Computer Science) aims to equip students to qualify for joining a profession or to provide development opportunities in particular employment settings. Graduates are enabled to enter a variety of jobs or to continue academic study at a higher level.

Qualification descriptors for this Postgraduate Education reflect in-depth and advanced knowledge and understanding of their subjects enriched by scholarship, research and current practice. These include critical awareness of contemporary issues and developments; critical skills, knowledge of professional responsibility, integrity and ethos. Thus, qualification descriptor sets out the broad level of skills and competencies that Master's students are expected to achieve. They include generic information about what all holders of the qualification are able to do, and the qualities and skills that they have. These reflect student's different aspirations, motivations, learning needs and personal circumstances. Programmes assess not only academic skills but also other skills and attributes including what any professional body requires, recognises and accredits the award of Master's Degrees.

The characteristics associated with the specialised study such as M.Sc.(Computer Science) predominantly composed of structured learning opportunities. This programme is devoted to research project, leading to dissertation. Training in latest technologies is involved in this programme of study. Students are likely to be further characterized by their ability to study independently, and to use a range of research methods and techniques applicable to advance scholarship in the subject. The ability to complete a research in the subject includes a critical review of existing IT platforms or other scholarly outputs. They are able to apply research and critical perspective to professional situations both practical and theoretical.

4.1. Aims of Master of Science Degree Programme in Computer Science

It aims to provide students with a rigorous and integrated academic study of Computer Science. Students who complete the Master of Science Degree Programme in Computer Science successfully should:

- acquire an understanding of the principles of various IT platforms and Computer Science;
- acquire an improved ability to think analytically about different subjects in Computer Science and to apply this knowledge in their professional and national settings;
- acquire an improved ability to conduct research in the field of Computer Science;
- institutionalise framework for cross-national professional collaboration and the exchange of information;
- communicate their conclusions clearly;
- demonstrate self-direction and originality in tackling and solving problems, and in collecting and commenting on complex information;
- indicate ways of extending practices in Computer Science and apply various technologies to quickly evolving situations

4.2. Postgraduate Attributes in Computer Science

The postgraduate attributes in Computer Science involve skills expected to be gained by a student through studies that support in sharpening competence for augmenting contemporary knowledge base, acquiring new learning and skills, identifying with future studies, engaging well in a preferred career and performing a positive role as enlightened citizen in the society. The characteristic, profundity and magnitude of the learning experiences made available to the students support them to unfold the quality attributes in the following manner:

- **Disciplinary Knowledge:** Aptitude to manifest wide and extensive knowledge in the field of study and comprehension of one or more disciplines constitute part of postgraduate attributes including how other disciplines relate to the field of knowledge. An international perspective in the area of study also gives a wider learning of the subject. In the specialised course on Computer Science, the constant review and renewal of subject and courses assure coverage of recent developments. Quality education and training build a condition in which learning is exchanged, critically evaluated and used in contemporary situations with the aptitude to review, examine and integrate and utilize actual learning in the appropriate field.

- **Communication Skill:** Classroom discussion and formal presentations render a suitable opportunity to sharpen oral communication and written assessment skills. They create ability to manifest ideas and thoughts in writing and orally to communicate confidently their viewpoints. By expressing adeptness to listen meticulously, they can read and write logically as well as give obscure information in explicit and succinct manner. With practice as a part of interdisciplinary team, students become able to choose and employ the proper form and methods of communication.

- **Critical Thinking:** The ability to apply critical reasoning to issues through independent thought and informed judgment are important postgraduate attributes integrating information from a wide range of sources. The postgraduates are able to apply analytical thought to body of knowledge and critically evaluate ideas, arguments, claims, beliefs on the basis of empirical evidence from open-ended and reasoned perspectives. They become able to identify relevant assumptions or implications and formulate coherent arguments.

- **Research Related Skills:** Research papers and other research tasks are expected to develop a degree of creativity, originality and discovery that benefits a postgraduate programme of the highest quality and to which students are encouraged. An ability is developed to undertake supervised research, including the design and conduct of investigations in a systematic, critical manner. Identification of appropriate problem and research questions, a critical analysis of the literatures, drawing logical conclusion are integral part of research skills. Postgraduate programme in Computer Science is designed to enhance skills in research and analysis, which are tested in all forms of assessment. All postgraduates demonstrate, through subject assessment, their ability to develop substantial research-based scholarship. Research related skill involves a sense of inquiry and capability for asking relevant questions, defining problems, articulation, ability to recognise cause and effect relationship, formulate hypothesis, and to report the result of experiment or investigation.

- **Self-Directed Learning:** The demanding nature of postgraduate attributes requires effective time-management and an ability to work independently. The rigour of

programmes ensure that all postgraduates have developed the ability to work with relative autonomy, which provides a foundation for future leadership roles. Ability to work and learn independently and effectively leads to generating innovative ideas in the changing environment to investigate problems and to have creative solution. Self-learning and application of competence in exploring also help in solving non-familiar problems. This leads to application of one's learning to real life situation and critical sensibility to lived experiences. Well-developed problem-solving abilities also contribute to flexibility of approach.

- **Ethical and Social Understanding:** Profound respect for truth and intellectual integrity including the ethics of scholarship add to the ability to embrace values in conducting one's life and in formulating position about ethical problems from multiple perspectives appreciating environmental and sustainability issues. This postgraduate attribute fosters understanding of social and ethical responsibility and ability to apply ethical standards in order to attain unbiased and truthful actions in all aspects of life. It also involves appreciation of the philosophical and social contexts of a discipline with knowledge of other cultures and appreciation of cultural diversity.
- **Quality of Teamwork:** Teamwork, as postgraduate attributes, creates capacity to value and work effectively and respectfully with diverse team and to facilitate coordinated effort for a common cause. It involves training in mapping out tasks of a team, setting directions and formulating an inspiring vision.

4.3. Qualification Descriptors

The qualification descriptors indicate both disciplinary knowledge and understanding as well as generic skills, including global competencies that all students in postgraduate programmes of study for the award of qualification of M.Sc. Degree in Computer Science should demonstrate.

The students, who complete the course successfully for the Master's Degree in the subject, acquire an understanding of the principles and institutions of Computer Science. The qualification descriptors reflect an improved ability to think analytically about the concept, implementation and development of Computer Science their own professional and national settings. These descriptors also describe an improved ability to conduct research Computer Science in the institutional framework for national or cross-national professional collaboration and the exchange of information.

Postgraduates will have:

- an advanced and integrated knowledge of Computer Science for the protection and promotion of Computer Science;
- an advanced appreciation of the relationship between Computer Science and society, at the international and domestic levels, in the field of Computer Science;
- the cognitive and technical skills to independently examine and critically evaluate current issues by reference to international Computer Science standards.

Further, the postgraduates will also be able to:

- understand and critically examine the interrelationship between international, regional and domestic histories, philosophies, policies and practices of Computer Science

- engage as informed and open-minded participant in debates about Computer Science and its application;
- analyze, interpret and assess the challenges posed to Computer Science in the context of globalization; and
- demonstrate autonomy, expert judgment and responsibility as advocate in the field of Computer Science.

The students who complete the postgraduate programme of study will be awarded a Master's Degree in Computer Science discipline. Some of the qualification descriptors a postgraduate will be capable to demonstrate on completion of Master level programme will include the following:

- systematic, extensive, coherent knowledge and understanding of Computer Science study as a whole with its links to related disciplinary areas; critical comprehension of theories, principles and concepts; and understanding of emerging issues in Computer Science ;
- procedural knowledge related to the study of Computer Science, including research and development;
- skills in one's specialization and contemporary developments in Computer Science study, including critical understanding of latest developments in Computer Science;
- comprehensive knowledge about current research and skills for identifying problem relating to Computer Science study; analysis and interpretation of data using methodologies for formulating evidence based solutions and argument; and skill for critical assessment of wide range of ideas and complex problems relating to Computer Science;
- application of disciplinary knowledge and skills to unfamiliar context with ability to analyse issues and seek solution to real-life problem; and
- Computer Science related skills to job trades and employment opportunities.

4.4. Programme Specific Learning Outcomes M.Sc. (Computer Science**)**

Programme Learning Outcomes in Computer Science course include subject-specific skills and generic skills, including transferable global skills and competencies, the achievement of which students are able to demonstrate for the award of Masters Degree; M.Sc.(Computer Science). It is to develop expertise to:

- explore the conditions and dimensions of empowering and transformative learning processes;
- provide an advanced qualification for students wanting to better understand the nature of international human rights in the face of global political, economic, social, legal, ethical and environmental challenges;
- describe and critique the differing approaches, perspectives, and models of human rights and how they impact the ways in which human rights education is carried out in diverse settings;
- design, conduct, analyze and present findings using diverse research tools and methods in order to create knowledge and awareness about Computer Science;
- identify diverse methodological tools and skills needed to conduct ethical research;

- synthesize contextual understanding, reflective analysis, theoretical frameworks, and methodological training to inform the production of a thesis/project report and field-based research projects;
- demonstrate the aptitude of Computer Programming and Computer based problem-solving skills
- display the knowledge of appropriate theory, practices and tools for the specification, design, implementation
- ability to learn and acquire knowledge through online courses available at different MOOC Providers
- ability to link knowledge of Computer Science with other two chosen auxiliary disciplines of study
- display ethical code of conduct in usage of Internet and Cyber systems
- ability to formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate
- ability to operate, manage, deploy, configure computer network, hardware, software operation of an organization
- ensure comparability of learning levels and academic standard across universities;
- focus on knowledge and skill for further study, empowerment and citizenship;

4. Course Structure

The M.Sc. (Computer Science) course will be two-year full time course consisting of minimum four semesters and with a minimum of 94 credits. The medium of instruction and examination will be only English. The credit allotment for M.Sc. (Computer Science) course: Semester I (24 Credits), Semester II (26Credits), Semester III (26 Credits), and Semester IV (18 Credits). In each Semester, there will be four papers(three core compulsory and one core elective) of 100 marks each, two Laboratory course and minor project for each Semester of 100 marks each, out of which 40 marks will be for internal assessment and 60 marks for university examination. Fourth semester is internship for 200 marks. Thus M.Sc. (Computer Science) degree examination, four semesters shall be 2400 marks and of minimum 94credits altogether. The following shall be the course structure:

SEMESTER-WISE COURSE INFORMATION
SEMESTER I

Semester	Subject Type	Code	Title of the paper	Hrs/ Week	Credits	Exam Hrs	Maximum Marks		
							Internal Assessment	University Examination	Total
Semester I	Core: Compulsory	PGCS-101	Algorithm Design Patterns	04	04	03	40	60	100
		PGCS-102	Paradigm of programming Languages	04	04	03	40	60	100
		PGCS-103	Advanced Database Concepts	04	04	03	40	60	100
		PGCS-MI	Minor Project –I	04	04	03	40	60	100
	Core: Elective	**Any one from the following:**							
		PGCS-104	Parallel Processing	04	04	03	40	60	100
		PGCS-105	Theory of Automata	04	04	03	40	60	100
		PGCS-106	Digital Image Processing	04	04	03	40	60	100
		Any two from the following:							
	Elective : Practical's	PGCS-107	Lab Course –I	04	02	03	40	60	100
		PGCS-108	Lab Course –II	04	02	03	40	60	100
		PGCS-109	Lab Course –III	04	02	03	40	60	100
			TOTAL	24					

SEMESTER II

Semester	Subject Type	Code	Title of the paper	Hrs/Week	Credits	Exam Hrs	Maximum Marks		
							Internal Assessment	University Examination	Total
Semester II	Core: Compulsory	PGCS-201	Software Project Management	04	04	03	40	60	100
		PGCS-202	Cloud Computing	04	04	03	40	60	100
		PGCS-203	Java Application Programming	04	04	03	40	60	100
		PGCS-MII	Minor Project -II	04	04	03	40	60	100
	Core: Elective	**Any one from the following:**							
		PGCS-204	Network Security	04	04	03	40	60	100
		PGCS-205	Embedded Computing	04	04	03	40	60	100
		PGCS-206	Data Mining	04	04	03	40	60	100
		Any two from the following:							
	Elective Practical's	PGCS-207	Lab Course –IV	04	02	03	40	60	100
		PGCS-208	Lab Course –V	04	02	03	40	60	100
		PGCS-209	Lab Course –VI	04	02	03	40	60	100
	Ability Enhance-ment Course	PGAEC 11	Soft Skills	02	02	02	20	30	50
			Total		26				

SEMESTER III

Semester	Subject Type	Code	Title of the paper	Hrs/Week	Credits	Exam Hrs	Maximum Marks		
							Internal Assessment	University Examination	Total
Semester III	Core: Compulsory	PGCS-301	Artificial Intelligence	04	04	03	40	60	100
		PGCS-302	Mobile Technologies	04	04	03	40	60	100
		PGCS-303	.Net Technologies	04	04	03	40	60	100
		PGCS-MIII	Minor Project –III	04	04	03	40	60	100
	Core: Elective	**Any one from the following:**							
		PGCS-304	Software Architecture	04	04	03	40	60	100
		PGCS-305	Software Testing	04	04	03	40	60	100
		PGCS-306	Advanced Operating System	04	04	03	40	60	100
	Elective Practical's	**Any two from the following:**							
		PGCS-307	Lab Course –VII	04	02	03	40	60	100
		PGCS-308	Lab Course –VIII	04	02	03	40	60	100
		PGCS-309	Lab Course –IX	04	02	03	40	60	100
	Skill Enhance-ment Course	PGSEC 31	Android Programming	02	02	02	20	30	50
			Total		26				

SEMESTER IV

Semester	Code	Title of the course	Credits	Maximum Marks		
				Internal Assessment	University Examination	Total
Semester IV	PGCS-401	Internship	18	80	120	200

5. Scheme of Examination:

The Assessment of Regular students of Master of Computer Science, M.Sc.(Computer Science)course in the academic session 2018-19 and after, shall be based on

(a) University Examinations,
(b) Internal Assessment,
(c) Choice Based Credit System, and
(d) Semester Grade Point Average and Cumulative Grade Point Average system

For each paper of 100 marks, there will be Internal Assessment of 40 marks and the University Examination of 60 marks/3 hours duration at the end of each semester. The 04 credit will be given to a student who secures at least 40% of marks allotted to each paper. A candidate who does not pass the examination in any subject or subjects in one semester will be permitted to reappear in such failed subject or subjects along with the papers of following semesters.

The Internal Assessment (IA) for each paper will be of 40 marks which will be carried out by the department during the term. The Internal Assessment may be in the forms as follows: Attendance, Written tests, seminars, term papers, presentations, assignments, orals or any such others. There will be at least two types of assessments from the types given above.

At the end of each semester, a cumulative grade point average (CGPA) and also Semester grade point average(SGPA) will be calculated as a weighted average of the GPI of all courses in which the student has passed till that semester.

A candidate shall be permitted to proceed from the First Semester up to Final Semester irrespective of his/her failure in any of the Semester examinations subject to the condition that the candidates should register for all the arrear subjects of earlier semesters along with current (subsequent) semester subjects.

Practical and Project Work:

Each practical examination for laboratory course is of 100 marks and three hours duration. The minor projects in Semesters I ,II and III will be evaluated for 100 marks for the allotted credits by a panel consisting of one internal and one external examiner .For both laboratory course and minor project , there will be internal assessment of 40 marks and the university examination of 60 marks. The project work is to be undertaken under guidance of a teacher allotted to a student by the department.

The candidate has to submit the project report before the deadline announced by the department. A candidate who fails to submit the project may resubmit the same in the subsequent semester examination for evaluation. The project work activities must be duly supported by documentary evidences to be endorsed by the Head or the Guide.

6. Scheme of credits:

The M.Sc. (Comp. Sci.) is of 94credits. The distribution of credits over semesters is given below.

Semester	Total credits
Sem I	24
Sem II	26
Sem III	26
Sem IV	18
Total	94

7. Standard of Passing:

For all courses, both UE and IA constitute separate heads of passing. In order to pass in such courses and to earn the assigned credits, a student must obtain a minimum grade point of 5.0 (40% marks) at UE and also a minimum grade point of 5.0 (40% marks) at IA.

Even a student fails in IA, he/she shall be declared 'pass' in the course provided he/she obtains a minimum of 25% in IA and GPA for the course is at least 6.0 (50% in aggregate). The GPA for a course will be calculated only if the student passes at the UE.

A student who fails at UE in a course has to reappear only at UE as a backlog candidate and clear the head of passing. Similarly, a student who fails in a course at IA has to reappear only at IA as a backlog candidate and clear the head of passing.

The 10-point scale Grades and Grade Points according to the following table.

Range of Marks (Out of 100)	Grade	Grade Point
80 ≤ Marks ≤ 100	O	10
70 ≤ Marks < 80	A+	9
60 ≤ Marks < 70	A	8
55 ≤ Marks < 60	B+	7
50 ≤ Marks < 55	B	6
40 ≤ Marks < 50	C	5
Marks < 40	D	0

The performances at UE and IA will be combined to obtain the Grade Point Average (GPA) for the course. The weights for performance at UE and IA shall respectively be 60% and 40%.

GPA is calculated by adding the UE marks out of 60 and IA marks out of 40. The total marks out of 100 are converted to grade point, which will be the GPA.

8. Formula to Calculate Grade Points (GP):

Suppose that '*Max*' is the maximum marks assigned for an examination or evaluation based on which GP will be computed. In order to determine the GP, Set $x = Max/10$ (since we have adapted 10-point system). Then GP is calculated by the formulas shown as below.

Range of Marks at the evaluation	Formula for the Grade Point
$8x \leq$ Marks $\leq 10x$	10
$5.5x \leq$ Marks $< 8x$	Truncate (Marks/x) +2
$4x \leq$ Marks $< 5.5x$	Truncate (Marks/x) +1

Two kinds of performance indicators, namely, the Semester Grade Point Average (SGPA) and the Cumulative Grade Point Average (CGPA) shall be computed at the end of each term. The SGPA measures the cumulative performance of a student in all the courses in a particular semester, while the CGPA measures the cumulative performance in all courses since his/her enrolment to the course. The CGPA of a student when he/she completes the programme is the final result of the student.

The SGPA is calculated by the formula SGPA = $\frac{\sum Ck \times GPk}{\sum Ck}$, where C_k is the credit-value assigned to a course and GP_k is the GPA obtained by the student in the course. In the above, the sum is taken over all the courses that the student has undertaken for the study during the semester, including those in

which he/she might have failed or those for which he/ she remained absent. The SGPA shall be calculated up to two decimal place accuracy.

The CGPA is calculated by the formula CGPA = $\frac{\sum Ck \times GPk}{\sum Ck}$, where C_k is the credit-value assigned to a course and GP_k is the GPA obtained by the student in the course. In the above, the sum is taken over all the courses that the student has undertaken for the study from the time of his/her enrolment to the course and also the during the semester for which CGPA is calculated, including those in which he/she might have failed or those for which he/she remained absent. The CGPA shall be calculated up to two decimal place accuracy.

The Formula to compute equivalent percentage marks for specified CGPA:

% Marks (CGPA) =	10 × CGPA – 10	if 5.00 ≤ CGPA ≤ 6.00
	5 × CGPA+ 20	if 6.00 ≤ CGPA ≤ 8.00
	10 × CGPA – 20	if 8.00 ≤ CGPA ≤ 9.00
	20 × CGPA – 110	if 9.00 ≤ CGPA ≤ 9.50
	40 × CGPA – 300	if 9.50 ≤ CGPA ≤ 10.00

9. Award of honours:

A student who has completed the minimum credits specified for the programme shall be declared to have passed in the programme. The final result will be in terms of letter grade only and is based on the CGPA of all courses studied and passed. The criteria for the award of honours are given below.

Range of CGPA	Final Grade	Performance Descriptor	Equivalent Range of Marks (%)
9.50 ≤CGPA≤ 10.00	O	Outstanding	80 ≤ Marks ≤ 100
9.00 ≤CGPA≤ 9.49	A+	Excellent	70 ≤ Marks < 80
8.00 ≤CGPA≤ 8.99	A	Very Good	60 ≤ Marks < 70
7.00 ≤CGPA≤ 7.99	B+	Good	55 ≤ Marks < 60
6.00 ≤CGPA≤ 6.99	B	Average	50 ≤ Marks < 55
5.00 ≤CGPA≤ 5.99	C	Satisfactory	40 ≤ Marks < 50
CGPA Below 5.00	F	Fail	Marks Below 40

A candidate shall be permitted to proceed further from the First Semester up to Fourth Semester irrespective of his/her failure in any of the Semester examinations subject to the condition that the candidates should register for all the backlog subjects of earlier semesters along with current (subsequent) semester subjects.

10. Gracing:

The gracing shall be done as per existing rules of the University.

11. Verification and Revaluation:

There is provision for verification and revaluation of the result. A student can apply for the verification and revaluation of the result within the two weeks from the declaration of the results with the prescribed fee. The verification and revaluation shall be done as per the existing rules of the University.

12. Format of the transcript:

The student will be given a transcript indicating his/her performance at the end of every semester examination. The transcript shall be given as per the following table along with other necessary details:

Course No.	Course Name	No. of Credits	University Examination		Internal Assessment		Grade Point Average	Result
			Grade	Grade Point	Grade	Grade Point		
1								
2								
3								
4								
5								
Total Cumulative Credits completed			SGPA		CGPA		Equivalent Marks (%)	
Note: GPA is calculated by adding the UE marks out of 60 and IA marks out of 40. The total marks out of 100 are converted to Grade Point, which will be the GPA.								

* * *

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS-101: Algorithm Design Patterns

Course Outcomes:

At the end of the course, a student shall be able to:

- understand different algorithm design techniques and their features
- create efficient algorithms for real world problems
- implement techniques for synthesis & analysis of algorithms

Total Credits: 04 **Total lectures: 60**

Course content

1 .Introduction: (4L)

Algorithm Concept, Writing structured programs, Analyzing algorithms

2. Divide & conquer: (10L)

The General method, Binary search, Finding the maximum &minimum selection, merge sort

3. The greedy method: (12L)

The general method, Optimal storage on tapes, Knapsack problem, Job Sequencing with deadlines, Optimal merge patterns, Minimum spanning tree

4. Dynamic programming: (12L)

The general method, Multistage Graphs, All pairs shortest path, Optimal binary search tree, Flow shop scheduling

5. Basic search & traversal techniques: (12L)

The techniques, code optimization, AND/OR graphs Game tree, Backtracking. The general method, The 8-queens problems, Sun of subsets, Graph coloring, Hamilton cycles

6. Branch-and-Bound: (6L)

Strategy ,The method 0/1 knapsack problem, Traveling salesperson, LIFOBB, FIFOBB

7. NP-HARD&NP- COMPLETE problem:

(4L)

Basic concepts, cook's theorem, NP-HARD graph problem, NP-HARD scheduling problem

References Books:-

1) Fundamentals of computer Algorithms – Coreman
2) Algorithm Design – Kleinberg and Tardos

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I
PGCS-102: Paradigms of Programming Languages

Course Outcomes:

At the end of the course, a student shall be able to:

- think about programming languages analytically andseparate syntax from semantics.
- develop a greater understanding of the issues involved in programming language design and implementation.
- familiar with design issues of object – oriented and functional languages.
- analyze semantic issues associated with function implementations, including variable binding, scoping rules, parameter passing.

Total Credits: 04 **Total lectures: 60**

Course content

1. **Introduction to Programming languages: (6L)**
   Need for studying programming languages, A Short of programming languages, Features of a good language, Effects of environments on languages
2. **Principles of languages: (10L)**
   Design Structure and operation of a computer Virtual computers and languages
   Implementations syntax, type and semantics, Context free grammars, Grammar For Expression, Lexical analysis, parsing
3. **Paradigms of Programming language**: **(10L)**
   Imperative paradigms (simple Procedural languages) Fortran, C. Block structured paradigm Pascal. The object based paradigm-Ada, C++, small talk, Logic programming paradigm-Prolog, The database language paradigm, declarative paradigm, Event driven programming, Fourth generation languages
4. **Building blocks of a language : (8L)**
   Data Object, variable constants, Data types- A brief classifications, Derived, Abstract, User- Defined Type conversion
5. **Procedures: (8L)**
   Designing of procedure, Simple call return sub-program, Sub-programs, subroutines, Recursive subprogram, Referencing environment-local And global Different parameter passing call-By-value, call-by-reference
6. **Functional programming: (10L)**
   Characteristics of functional Programming, Elements of Functional Programming language, Functions, Functional forms, Functional declarations , Expression Evaluation Logic programming, Proofs, facts, queries, logical Variable, substitutions Instances.
7. **Object oriented programming: (10L)**
   Introduction Basic concepts-object, class, Characteristics of OOP, Abstraction, Encapsulation and information, Hiding, Inheritance, Multiple inheritance, Polymorphism, OOP concepts in C++
8. **Comparative study of programming languages: (4L)**
   Control structures Flow control, rules, Basic control structures

Reference Books :

1. Programming languages design $ implementation by terrace W.Pratt, Marvin V Zelkowit
2. Programming paradigms Methology by G.P.Potdar

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS-103: Advanced Database Concepts

Course Outcomes:

At the end of the course, a student shall be able to:

- translate complex conceptual data models into logical and physical database designs.
- design high-quality relational databases and database applications.
- understand location, replication and fragmentation independence
- implement advance database concepts and techniques

Total Credits: 04 **Total lectures: 60**

Course content

Object Oriented Databases:- **(12L)**

Overview of Object Oriented Concept ,Object Identity ,Object Structure , Object Definition Language, Types of Constructors , object database conceptual design, OODBMS advantages, Object query language examples of OODBMS

Distributed Databases : **(12L)**

Introduction to Distributed data processing, Homogeneous and heterogeneous systems ,Distributed DBMS Architecture, fragmentation , Distributed database design , Overview of Query processing, Query decomposition and data localization

Distributed DBMS reliability: **(12L)**

Reliability concepts & measures, Failures & fault tolerance in distributed systems, Local reliability protocols, Distributed reliability protocols, Network partitioning

Parallel Databases: **(12L)**

Parallel database concepts, Parallel database system architecture, Query parallelism, parallel Data processing and parallel query optimization

Emerging Database Technologies & Applications: **(12L)**

Multimedia Databases, Mobile Databases, Geographical Information Systems, Spatio – temporal patterns, Intervals and scalar operators in temporal databases, web databases,
Deductive Databases

Reference Books:-

1. Fundamentals of Database Management Systems By Navathe & Elmasri (3rd Edition, Pearson Education)
2. Principles of Distributed Database Systems by Patrick Valduriez (3rd Edition' Pearson Education)
3. Introduction to Database Systems by C G Date (7th Edition Pearson Education)

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – MI: Minor Project-I

Learning outcomes:

At the end of the course, a student shall be able to:

- apply knowledge and techniques learnt in theoretical classes for developing the s/w for real problems.
- Get an insight into the working of the real organizations/companies.
- gain deeper understanding in specific functional areas.
- exploring career opportunities in their areas of interest.

Total Credits: 04

Course content

The course Minor Project is one that involves requirement analysis, feasibility analysis, Database design, coding, testing, implementation and maintenance.

Student will select individually Commercial or Technical project based on Technologies learnt in Semester I. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation. The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

- The Project can be platform, Language and technology independent.
- Project will be evaluated by project guide.
- Assessment will be done weekly in the respective batch.
- Evaluation will be on the basis of weekly progress of project work, progress report, oral, results and documentation and demonstration.
- You should fill your status of the project work on the progress report and get the Signature of project guide regularly.
- Progress report should sharply focus how much time you have spent on specific task. (The format of progress report is given as follow.)
- You should keep all signed progress reports.
- Project will not be accepted if progress report is not submitted and all responsibility remains with student.
- Students should prepare design document using SE/UML techniques depends on your project.

About project Report:

The report should be typed on A4 size, executive bond paper for the final submission. The report should be in the good quality Rexene bound. We suggest, using one-and-half spaced printing, Times New Roman 12 font sizes for the normal text, 14-16 font sizes for headings & page titles. Number of copies: For one project you should prepare 2 copies of the project report. One for yourself, one for college (College copy can be in CD).

Evaluation for internal 40 marks

Description	Marks
UML /ERD/DFD diagrams	10
Technology and design base first demo	10
Project technology based two assignments	10
Second Demo	10

Evaluation for external 60 marks

Description	Marks
Demo	10
Report	10
Presentation	20
Viva	20

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

Core Elective-I PGCS-104: Parallel Processing

Course Outcomes:

At the end of this course, a student shall be able to :

- understand the fundamental aspects of parallel processing,
- familiar with taxonomies of parallel system,
- familiar with performance measures for parallel system,
- understand the theoretical limitations of parallel computing,
- write efficient parallel application programs.

Total Credits: 04 **Total lectures: 60**

Course content

1. Overview:
Motivation of parallelism, Scope of parallel Computing, Current trends in parallel processing **(4L)**

2. Concept Parallel Machine Model: **(6L)**
Parallel Programming Model, Parallel algorithm examples

3. Designing Parallel Algorithms: **(4L)**
Methodical Design, Partitioning, Communication, Agglomeration, Mapping, Case Studies

4. Quantitative basis for design: **(4L)**
Defining & Modeling performance, Developing Models, Scalability analysis, Communication cost model, Case Studies

5. Modular Designs: Modularity & parallel computing Case studies **(4L)**

6 .Tools: **(12L)**
Compositional C++ ,C++ review ,C C++: Introduction C++: Communication Case studies, Fortran M, Introduction, Communication, Argument passing, mapping, modularity, performance issues ,High performance Fortran, Data parallelism, data distribution & concurrency, Modularity, other HPF features & performance issues

7. Message passing interface: **(8L)**
MPI models & MPI basics, Communication, modularity & other features

8. Performance tools : Performance analysis, Tools related to performance analysis **(6L)**

10.Hypercube algorithm: Vector reduction & matrix transposition, Merge sort **(4L)**

11.Summary & revision: Parallel & distributed computing, Programming & analytical tools **(4L)**

Textbooks:

1. Designing &building parallel programs (2004) by I. Foster. Addison Wesley, ISBN 0-201-57594-9
2. Introduction to Parallel Computing, 2nd ed.(2003) by Ananth Grama, Anshul Gupta, George Karypis & Vipin Kumar. Benjamin/Cummings, ISSBN 0-8053-3170-0.

Reference Books:

3. Introduction to Parallel Processing by Prakash P. Ravi, Sasikumar M. & Shikhare Dinesh. Prentice Hall of India
4. Practical parallel programming by Wilson Gregory V. Prentice Hall of India
5. Parallel computer architecture (1999) by D.Culler & J.P.Singh. Morgan Kaufmann
6. Distributed & parallel computing (1997) by h. El-Rewini & t.g.Lewis
7. parallel programming with MPI(1996) by peter Pacheco Barnes & noble

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I
Core Elective-I PGCS-105: Theory of Automata

Course Outcomes:

At the end of this course, a student shall be able to :

- understand the core concepts in automata theory and formal languages.
- design grammars and automata (recognizers) for different language classes.
- identify formal language classes and prove language membership properties.
- to prove and disprove theorems establishing key properties of formal languages and automata.
- understand core concepts relating to the theory of computation and computational models

Total Credits: 04 **Total lectures: 60**

Course content

1. Introduction: Review of Mathematical, Preliminaries, Relations, Functions, Set theory, and Predicate and Prepositional, Calculus, principle of Mathematical induction /strong, Mathematical induction **(4L)**

2. Finite state machine: Definition, finite control, transition graphs, adjacency matrix, finite automata, determinist finite automata, language acceptance by FA, Moore and Mealy Machines, Finite state machine with output, FA MINIMIZATION AND RELATED THEOREM. **(10L)**

3. Regular Expression: Recursive regular expression, regular set, NFA with E moves, NFA without E moves, Inter conversion between NFA and DFA, Regular expression and FA, pumping LEMMA **(8L)**

4. Grammar: Context free grammar and it's properties, derivation tree, simplifying CFG, unambiguous CFG, CNF, GNF of CFG, Chomsky hierarchy, Chomsky normal form, derivation graphs type 0 and type 1 grammar. Concept of linear bounded automata, context sensitive grammars and their equivalence **(8L)**

5. Push down automata: Push down automata 2 way PDA, relation of PDA with CFG, deterministic and nondeterministic PDA and related theorems **(6L)**

6. Turing Machine: Definition model, comparison of Turing machine, example of TM, universal TM, tm limitation, church's Turing hypothesis ,multi stack TM ,halting problem, unrestricted grammars and their equivalence with TM, determinism and non-determinism of TM, TM as acceptor /generator ,algorithms and related theorems multi-tape, multi-head, multi-stack TM **(12L)**

7. Complexity: Introduction recursively numerable, sets, recursive set, partial recursive sets, Russell paradox undesirability and some non-computable problems **(8L)**

8. Applications: Application of RE and PA – lexical analyzer, text editor and searching using RE, application of PDA expression conversion, application on CFG, syntax analysis **(8L)**

Reference Books:

1. Daniel LA Cohen, "Introduction to computer theory". Wiley Publication.
2. John C. Martin, "Introduction to language and theory of computation", Mc Grawhill.
3. HopCroft Ullman,"Introduction to automata Theory, Language and Computations", Narosa.
4. Hapcraft Ulman "Introduction to Automata Theory".
5. Harry R. Lewis,"Elements of Thoery Of Computation"
6. E.V. Krishnamurthy "Theory of computer science" EWP publications.
7. LIU.C.L., Elements of Discrete Mathmatics, Mc Grawhill.
8. Aho,Ulman,Sethi "Principles of compiler construction"

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

Core Elective-I PGCS-106: **Digital Image Processing**

Course Outcomes:

At the end of the course, a student shall be able to:

- analyze general terminology of digital image processing.
- examine various types of images, intensity transformations and spatial filtering.
- develop Fourier transform for image processing in frequency domain.
- evaluate the methodologies for image segmentation, restoration etc.
- implement image process and analysis algorithms.
- learn different feature extraction techniques for image analysis and recognition

Total Credits: 04 **Total lectures: 60**

Course content

1. **What is Digital Image Processing: (4L)**

Low level image processing, High level image processing, The origins of Digital Image Processing, Examples of fields that use Digital Image Processing, X-Ray Imaging, Imaging in the Ultraviolet Band, Visible and Infrared Band, Microwave Band, Radio Band, Fundamental steps in Digital Image Processing, Components/Elements of Digital Image Processing

2. **Elements of Visual perception: (6L)**

Structure of Human Eye, Image formation in the Eye, Brightness, Adaptation and Discrimination, Light and electromagnetic spectrum, Image sensing and acquisition, Image acquisition using a single sensor, Image acquisition using sensor strips ,Image acquisition using sensor array, A single image formation model, Image sampling and quantization, Basic concepts of sampling and quantization, Spatial and gray level resolution, Aliasing and moiré patterns, Zooming and shrinking of digital image, Some basic ,relationship between pixels, Neighbors of a pixel, Adjacency, Connectivity, Regions, Boundaries, Distance measures, Linear and non linear operations

3. **Image Enhancement in the Spatial domain**: **(10L)**

Introduction, Some basic grey level transformation Image negatives ,Log transformation, Power law(Gamma) transformations ,Piece wise linear transformation functions, Histogram processing, Histogram equalization ,Histogram matching(specification),Local enhancement, Image enhancement using arithmetic and logical operation, Image substation, Image averaging, Basics of spatial filtering, Smoothing spatial filters, Order statistic (non linear) filters, Sharpening spatial filters, Use of second derivative for enhancement-the Laplacian, Use of first derivatives for enhancement – the gradient, Combining spatial enhancement methods

4. **Image Enhancement in the Frequency domain: (10L)**

Introduction to the Fourier Transform and the Frequency Domain, One Dimensional Fourier Transform,2-D Discrete Fourier Transform (DFT)and its Inverse, Filtering in the Frequency Domain, Correspondence between Filtering in the Spatial and frequency domain, Smoothing Frequency –Domain Filters

5. **Image Restoration: (12L)**

Introduction, Noise Models, Gaussian Noise, Rayleigh Noise, Erlang Noise ,Exponential noise, Uniform noise, Impulse Noise, Periodic noise, Restoration in the presence of noise only spatial filtering, Periodic noise reduction by frequency domain filtering, Band reject alters, Bandpass filters, Notch filters, Estimating the Degradation Function, Estimation of

Degradation Function by Image Observation, Estimation of Degradation Function by Experimentation, Estimation of Image degradation by Modeling Geometric mean Filter, Inverse Filtering, Minimum Mean square error(Wiener)filtering ,Geometric Transformation

6. Morphological Image Processing: (8L)

Some basic concepts from set theory, Reflection and Translation, Logic operation involving Binary Images, Erosion and Dilation, Erosion, Dilation Duality, Opening and Closing, The Hit –or-Miss Transformation

7. Image Segmentation Fundamentals: (4L)

Detection of Discontinuities, Thresholding , Region –based Segmentation

8. Representation and Description: (6L)

Representation, Simple Boundary Descriptors, Simple Regional Descriptors, Use of principal components for description, Relational Descriptors

Reference books

1. Fundamental of Digital Image Processing by Anil k. Jain Prentice -Hall
2. Digital Image Processing by R.C. Gonzalez and Woods
3. Digital Image Processing using Matlab by R.C. Gonzalez and Woods,S .L Eddnins
4. Digital Image Processing PIKS Scientific Inside by William .K.Pratt ,WILEY India
5. Digital Image Processing and Computer Vision by Sonka ,Hlavac,Boyle, CENGAGE Learning

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – 107: Lab Course - I

Course Outcomes:

At the end of this course, a student shall be able to:

- understand and explain different data structures and their features
- explore various applications of arrays, stack and queues and linked list
- implement Btrees, AVL Trees, sorting and searching algorithms
- design and execute programs using linked representation of stack and queue

Total Credits: 02

Course content

List of' programming assignments to be executed in C / C++:

1. Represent sparse matrix using array and perform matrix addition or simple and fast transpose.
2. Represent polynomial as a circular linked list and write a menu driven program to perform addition, multiplication and evaluation.
3. Write a menu driven program to perform following operations on doubly linked list: Create, Insert, Delete and Display.
4. Create two singly or doubly linked lists, sort them after creation using pointer manipulation. Merge these two lists into one list without creating, a new node. Merged list should be a sorted one.
5. Write a program to create a generalized linked list and perform following operations copy, equivalence and depth.
6. Implement Stack as an abstract data type-using array or linked list. Use this ADT for expression conversion and evaluation.
7. Represent circular Queue using, array and write a pro-ram to perform following operations Insert, Delete, Finding front and rear element.
8. Creation of binary tree and perform recursive and non-recursive traversals.
9. Creation of binary inorder threaded tree and perform all three traversals.
10. Represent a given graph using adjacency list and perform DFS and BFS.
11. Represent a (given graph using adjacency list or array and find the shortest path using Dijkstra algorithm.
12. Represent a given graph using adjacency list or array and (generate a minimum spanning tree using Kruskal's and Prime's algorithms.
13. Implement binary search tree as an abstract data type.
14. Create a binary search tree and find height of a tree and print the leaf nodes.
15. Create a binary search tree, find its mirror image, Print original and mirror image using level wise printing.
16. Create a hash table and handle the collisions using linear probing with perform or without replacement.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – 108: Lab Course - II

Course Outcomes:

At the end of the course, a student shall be able to:

- understand and implement C ,C++ programs logic.
- Implement the procedure and object oriented programming.
- develop applications using java and C# programming.
- apply file handling application through C, java and C# programming Languages.

Total Credits: 02

Course content

Any 16 assignments on the following topics

1. Basic programming structure using C,C++,JAVA,C#
2. Control structures: If statements, Loop structure
3. Programs based on Arrays: 1D,2D
4. String Manipulation
5. Functions: built in functions and user defined functions
6. Inheritance: single inheritance, multiple, multilevel and hybrid
7. Polymorphism using C++ and JAVA
8. Constructor
9. File handling

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-I

PGCS – 109: Lab Course - III

Course outcomes:

At the end of this course, a student shall be able to:

- understand and enhance the fundamental concepts of SQL.
- explore the knowledge about SQL environment by performing DDL and DML operations.
- analyze test results of given experiments work with operations of SQL

Total Credits: 02

Course content

Note: My Access / MySQL may be used.

- **DDL Commands**
    - Create table, alter table, drop table
- **DML Commands**
    - Select , update, delete, insert statements
    - Condition specification using Boolean and comparison operators (and, or,
- not,=,<>,>,<,>=,<=)
    - Arithmetic operators and aggregate functions(Count, sum, avg, Min, Max)
    - Multiple table queries (join on different and same tables)
    - Nested select statements
    - Set manipulation using (any, in, contains, all, not in, not contains, exists, not
- exists, union, intersect, minus, etc.)
    - Categorization using group by……..having
    - Arranging using order
- Procedures
- PL/SQL block
- Triggers

Overall, 16 assignments should be conducted.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

PGCS-201: Software Project Management

Course outcomes:

At the end of this course, a student shall be able to:

- explore similarities and differences between IT projects and other types of projects.
- apply general project management competencies to IT projects.
- built the techniques and develop the documents related to IT project management.
- understand how to apply different life-cycle models t design IT projects.
- analyze the nature of projects that plan plan-driven and agile development
- Identify IT project risks and develop risk mitigation strategies.

Total Credits: 04 **Total lectures: 60**

Course content

1.Introduction to competencies: (12L)

Product development technique - management skills - product development life cycle - software development process and models - The SEI CMM - international organization for standardization

2. Domain process (12L)

Managing domain process project selection models-project portfolio management – financial process selecting a project team – goal and scope of software project - project planning - creating the work breakdown structure – approaches to building a WBS – project milestones – work packages-building a WBS for software

3. Software development: (12L)

Tasks and activities – software size and reuse estimating- the SEI CMM – problems and task - cost estimation - effort measures - COCOMO: A regret ion model – COCOMO II – SLIM: a mathematical model - organizational planning - project roles and skills needed.

4. Scheduling activities: (12L)

Project management recourse activities - organizational form and structure –software development dependences - brainstorming – scheduling fundamentals - PERT and CPM – leveling recourses assignment-ma the schedule to a real calendar – critical chain scheduling

5. Quality assurance (12L)

Quality requirements – the SEI CMM – guideline – challenges - quality function deployment – building the software quality assurance – plan –software configuration management: principals – requirements - planning and organizing – tools – benefits - legal issues in software - case study

References Books:

1. Robert T.Futrell, Donald F. Shafer, Linda I. Safer, “Quality Software Project Management”, Pearson Education, Asia, 2002.
2. Pankaj Jalote,”Software Project, Manegment in Practice”, Addison Wesley, 2002
3. Hughes,” Software Project Management,3/E”,Tata McGraw-Hill,2004

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

PGCS- 202: Cloud Computing

Course outcomes:

At the end of this course, a student shall be able to:

- explain the core issues of cloud computing such as security, privacy, and interoperability
- choose the appropriate technologies, algorithms, and approaches for the related issues
- analyze the functioning of different components involved in web services cloud platform
- identify problems and explain, analyze, and evaluate various cloud computing solutions

Total Credits: 04 **Total lectures: 60**

Course content

1. Cloud Introduction: **(12L)**

Cloud Computing Fundamentals: Cloud Computing definition, Types of cloud, Cloud services: Benefits and challenges of cloud computing, Evolution of Cloud Computing , usage scenarios and Applications , Business models around Cloud – Major Players in Cloud Computing - Issues in Cloud - Eucalyptus - Nimbus - Open Nebula, CloudSim.

2. Cloud Services And File System: **(12L)**

Types of Cloud services: Software as a Service - Platform as a Service – Infrastructure as a Service - Database as a Service - Monitoring as a Service – Communication as services. Service providers- Google App Engine, Amazon EC2, Microsoft Azure, Sales force. Introduction to MapReduce, GFS, HDFS, Hadoop Framework.

3. Collaborating With Cloud: **(12L)**

Collaborating on Calendars, Schedules and Task Management – Collaborating on Event Management, Contact Management, Project Management – Collaborating on Word Processing, Databases – Storing and Sharing Files- Collaborating via Web-Based Communication Tools – Evaluating Web Mail Services – Collaborating via Social Networks – Collaborating via Blogs and Wikis. 185 CS-Engg&Tech-SRM-2013

4. Virtualization For Cloud : **(12L)**

Need for Virtualization – Pros and cons of Virtualization – Types of Virtualization – System Vm, Process VM, Virtual Machine monitor – Virtual machine properties - Interpretation and binary translation, HLL VM - Hypervisors – Xen, KVM , VMWare, Virtual Box, Hyper-V.

5. Security, Standards, And Applications: **(12L)**

Security in Clouds: Cloud security challenges – Software as a Service Security, Common Standards: The Open Cloud Consortium – The Distributed management Task Force – Standards for application Developers – Standards for Messaging – Standards for Security, End user access to cloud computing, Mobile Internet devices and the cloud.

Reference Books :

1. Bloor R., Kanfman M., Halper F. Judith Hurwitz "Cloud Computing " Wiley India Edition,2010
2. John Rittinghouse & James Ransome, "Cloud Computing Implementation Management and Strategy", CRC Press, 2010
3. Antohy T Velte ,Cloud Computing : "A Practical Approach", McGraw Hill,2009
4. Michael Miller, Cloud Computing: "Web-Based Applications That Change the Way You Work and Collaborate Online", Que Publishing, August 2008.
5. James E Smith, Ravi Nair, "Virtual Machines", Morgan Kaufmann Publishers, 2006.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II
PGCS-203: Java Application Programming

Course outcomes:

At the end of this course, a student shall be able to:

- understand the structure and model of the Java programming language
- develop software in the Java programming language
- apply practical knowledge about servlets and JSP
- implement JDBC connectivity with servlets and JSP.
- apply RMI and improve the knowledge about EJBs
- create web sites by using JSP

Total Credits: 04 **Total lectures: 60**

Course content

Pre-requisite: Students hould be familiar with basic programming concepts, basic objects oriented programming concept with java

1.Networking : (8L)

Networking basics, Connecting to the server, sockets for client and server one example writing client and server, UDP Datagram socket

2.Java data Base : (10L)

JDBC Introduction, Types and drivers, Querying databases, JDBC Metadata, Creating prepared statements, Scrollable result sets, Transaction management, connecting pooling, save points and performing batch updates, distributed Transaction support

3 .Java remote Method Invocation(RMI): (12L)

introduction to distributed computing using RMI, RMI Architecture, Writing simple RMI application, Invoking remote object, Object serialization for remote parameters

4.Servlet and JSP: (10L)

Introduction and life cycle, Servlet API, Servlet and Thread safety, HTTP Redirect, Managing User state: code, Session Tracking, servlet and JDBS, servletside Includes, JSP Introduction $ JSP Directive, JSP Scripting elements JSP Standard Action, JSP Implicit Object

5.Enterprise Java Beans: (8L)

Introduction to EJB, Session beans, Entity beans(CMP/BMP)JNDI, One example development of enterprise, JavaBeans, Connecting EJB to web module, Enterprise application Design consideration, Application deployment

6.MVC Architecture and Struts Framework (12L)

Introduction to MVC architecture, Development of controller Servlet, Development of controller Servlet, Development of Model bean, Development of view components, Component integration and deployment, Introduction to struts, Struts application framework, struts controller model and view components, JSP Custom tag library, Struts and EJB.

Reference Books:

1. core java vol -II CAY R. Horstman and Gray cornel, Pearson Education
2. Java Server programming, wrox press
3. Inside Servlet, Dustin R. Callway, Pearson Education
4. Entrprise JavaBeans, Richard Monson Hafel, Oreilly Publication

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II
PGCS – MII : MINOR PROJECT-II

Course Outcomes:
At the end of this course, a student shall be able to:

- demonstrate a sound technical knowledge of their selected project topic.
- undertake problem identification, formulation and solution
- design engineering solutions to complex problems utilizing a systems approach.
- communicate with engineers and the community at large in written or oral forms.
- demonstrate the knowledge, skills and attitudes of a professional software developer.

Total Credits: 04
Course content

The course Minor Project is one that involves requirement analysis, feasibility analysis, Database design, coding, testing, implementation and maintenance.

Student will select individually Commercial or Technical project based on Technologies learnt in Semester I. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation. The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

- The Project can be platform, Language and technology independent.
- Project will be evaluated by project guide.
- Assessment will be done weekly in the respective batch.
- Evaluation will be on the basis of weekly progress of project work, progress report, oral, results and documentation and demonstration.
- You should fill your status of the project work on the progress report and get the Signature of project guide regularly.
- Progress report should sharply focus how much time you have spent on specific task. (The format of progress report is given as follow.)
- You should keep all signed progress report.
- Project will not be accepted if progress report is not submitted and all responsibility remains with student.
- Students should prepare design document using SE/UML techniques depends on your project.

About project Report: -
The report should be typed on A4 size, executive bond paper for the final submission. The report should be in the good quality Rexene bound. We suggest, using one-and-half spaced printing, Times New Roman 12 font sizes for the normal text, 14-16 font sizes for headings & page titles. Number of copies: For one project you should prepare 2 copies of the project report. One for yourself, one for college (College copy can be in CD).

Evaluation for internal 40 marks

Description	Marks
UML /ERD/DFD diagrams	10
Technology and design base first demo	10
Project technology based two assignments	10
Second Demo	10

Evaluation for external 60 marks

Description	Marks
Demo	10
Report	10
Presentation	20
Viva	20

* * * * * *

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

Core Elective-II PGCS- 204: Network Security

Course Outcomes:

At the end of this course, a student shall be able to:

- understand concepts of network security and cryptographic techniques
- design and analyze cryptographic techniques
- solve network security issues in real time applications
- develop an understanding of security policies (such as authentication, integrity and confidentiality)
- understand and solve Electronic Mail Security issues
- use firewalls

Total Credits: 04 **Total lectures: 60**

Course content

1. Introduction : (8L)

OSI Reference Model, IP, UDP, and TCP, Replicated Directory Services ,Packet Switching, Network Components, Destination: Ultimate and Next Hop. Address Structure, Active Vs Passive Attacks, Viruses, Worms, Trojan, Horses, Multilevel Model of Security

2.Introduction to Cryptography: (12L)

What is Cryptography, Breaking an Encryption scheme, Types of Cryptography, Functions, Secret Key Cryptography ,Public Key Cryptography, Hash Algorithms

3. Secrete Key Cryptography: (12L)

Generic block Encryption ,Data Encryption standard (DES),International Data Encryption Algorithm (IDEA),Advanced Encryption Standard (AES), Modes of Operation, Encrypting Large Message, Generating MAC's, Multiple Encrypting DES, Hashes and Message Digests an Introduction

4. Public Key Algorithm: (10L)

Modular Arithmetic, RSA Algorithm, RSA Algorithm, Diffe-Signature standard (DSS),Number Theory, AES and Elliptic Curves

5. Authentication System: (10L)

Password-Based Authentication, Address-based Authentication ,Cryptographic Authentication Protocols, Eavesdropping and Server Database Reading, Trusted Intermediaries, Session Key Establishment, Authentication of people, Security Handshake Pitfalls

6. Electronic Mail Security: (6L)

Distribution List, Security Services for Electronic mail, Privacy Authentication of sources, Message Integrity, Proof of Submission and Delivery, Message Flow Confidentiality, PEM and S/MIME,PGP (Preety and Good Privacy)

7. Firewalls: (4L)

Packet Filters, Application Level Gateway, Encryption Tunnels, Security System**,** NetWare V3,V5, DCE Security

Reference Books:

1. Charlie Kaufman, Radia Perlman, Mike Speciner, "NETWORK SECURITY-
2. Private Communication in a PUBLIC WORLD", 2nd Edition, Prentice Hall.
3. Edward Amoroso, "Fundamental of Computer Security Technology", Prentice Hall.
4. William Stallings, "Cryptography and Network Security-Principle and Practice ", Prentice Hall.
5. Marsall D.Adams, Suil Jajodia and Harold J. Podell, Eds, "Information Security –an Integration Collection of Essays", IEEE Computer Society Press.
6. William R. Cheswick and Steven M. Bellowin, "Firewalls and Internet Security, Repelling the Wily Hacker", Addision-Wesley.
7. Gunter Schafer.Security in Fixed and Wireless networks.John Willy & Sons.
8. J.Schiller."Mobile Communications",Second edition,Addison-Wesley,2003
9. Charles P.Pfleeger,"Security in Computing",Prentice Hall.
10. Warwick Ford,"Computer Communication Security",Prentice Hall.
11. William Stalling,"Network Security Essentials",Prentice Hall.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

Core Elective-II PGCS-205: Embedded Computing

Course Outcomes:

At the end of this course, a student shall be able to:

- Get the overview of embedded systems
- Understand the concepts and practices involved in the embedded computing domain.
- apply concepts of real time system and Emulation and Debugging techniques
- analyze and apply various applications of embedded system

Total Credits: 04 **Total lectures: 60**

Course content

1. **An overview of embedded computing : (6L)**
   Introduction to embedded systems, Categories of embedded systems, Requirements of embedded systems, Trends in embedded software Development Embedded Processors, Memories & Peripherals
   - Microcontrollers 8051
   - Discrete processors : 8-bit architecture, 16/32 bit CISC, RISC, DSP
   - Integrated processors : ARM RISC
   - Choosing a processor
   - Memory systems : types (SRAM, DRAM, FLASH), organization, access time, validating the contents of memory Basic peripherals : parallel ports, timers, clocks

2. **Applications of Embedded Systems: (12L)**
   Consumer electronics, Control systems and industrial Automation, Biomedical systems, Field instrumentation, Handheld computers, Data communication, Network information appliances, Telecommunications, Wireless communications

3. **Real time system concepts (12L)**
   - Critical section of code
   - Foreground/ background systems
   - Resource, shared resource
   - Multitasking, task, task switch
   - Kernel, scheduler, non-preemptive kernel, preemptive kernel
   - Reentrancy, round-robin scheduling
   - Task priority, static priority, dynamic priority, priority inversions, assigning task priorities
   - Mutual exclusion, deadlock, synchronization, event flags, intertask communication
   - Interrupts : latency, response, recovery, ISR processing time, NMI

4. **Writing software for embedded systems: (12L)**
   - The compilation process : compile, link, load Cross compilers
   - Run-time-libraries : processor dependent, I/O dependent, system calls, exit routines
   - Writing a library, using alternative libraries
   - Porting Kernels
   - C extensions for embedded systems

- Buffering and other data structures Linear buffers, Directional buffers, double buffer in, Buffer exchange, Linked lists, FIFO, Circular buffers, Buffer underrun and overrun, Allocating buffer memory, Buffer leakage

5. **Emulation and Debugging techniques** : **(6L)**
   - Debugging techniques: HLL simulation, low level simulation, on-board debugger, task level debugging, symbolic debug
   - Emulation
   - Optimization problems
6. **Basic design using RTOS** : **(4L)**
   - Principles
   - Encapsulating semaphores and queues
   - Hard real time scheduling considerations
   - Saving memory space
   - Saving power
7. **Real time without RTOS** : **(8L)**
   - Choosing the SW environment
   - Deriving real time performance from non-real time system
   - Scheduling and data sampling
   - Controlling from an external switch
   - Problems

Reference Books:

1. Dreamtech Software Team, 'Programming For Embedded Systems', Wiley-Dreamtech India Pvt.
2. Lewis D.W., 'Fundamentals of Embedded software: Where c and assembly Meet', Pearson Education Asia
3. Embedded Systems Design, 2e, Heath, Elsevier, ISBN:9788181479709
4. Embedded Systems Design with FPGAs, Saas, Elsevier, ISBN: 9789380501918
5. Programming Embedded Systems – Michael Barr
6. Embedded Systems Building Blocks _ Jean J. Labrosse
7. An Embedded Software Primer _ David E. Simon published by Pearson Educations

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

Core Elective-II PGCS-206: Data Mining

Course outcomes:

At the end of the course, a student shall be able to:

- understand various algorithms used for data mining
- analyze the data using existing data mining tools
- apply operations like association , classification and clustering for a given dataset

Total Credits: 04 **Total lectures: 60**

Course content

1. Data warehousing: (10L)

Need for data warehousing , architecture of DW, benefits of DW,OLAP and data cubes, data preprocessing – need, data cleaning, data integration and transformation, data reduction, roll up ,drill down.

2. Data marts: (4L)

Definition, reasons for creating data marts, designing data marts.

3. Introduction to Data Mining: (12L)

Definition, basic data mining tasks, knowledge discovery in databases, issues in the data mining, applications of data mining.

4. Data mining techniques: (8L)

Association rules, frequent item sets and association rule mining: apriori algorithm, FP growth algorithm.

5. Classification: (12L)

Definition, need of classification, decision tree learning, Bayesian classification, Naive Bayes classifier, linear classifiers, linear regression, and non linear regression.

6. Clustering: (8L)

Definition, need of clustering, types of clusters, similarities and distance measures, partitional algorithms like nearest neighbor algorithm- means.

7. Applications of data mining: (6L)

Social impacts of data mining, mining text database, mining spatial databases, mining web data.

Reference Books:-

1. Jiawei Han, micheline Kamber,"Data mining concepts and Techniques", Morgan Kaufmann publishers,2002.
2. Alex Berson,Stephen J.smith,"data warehousing, data mining and OLAP" Tata McGraw Hill 2004
3. M.H. Dunham "data mining" Pearson education.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

PGCS-207: Lab Course -IV

Course outcomes:

At the end of the course, a student shall be able to:

- define & implement Virtualization using different types of Hypervisors
- apply steps to perform on demand Application delivery
- examine the installation and configuration of Open stack cloud
- analyze and understand the functioning of different components involved in web services cloud platform
- implement functioning of Platform as a Service
- design & Synthesize Storage as a service using Cloud

Total Credits: 02

Course content

Software Lab based on Cloud Computing:

1. Create virtual machines that access different programs on same platform.

2. Create virtual machines that access different programs on different platforms.

3. Exploring Google cloud for the following **:**
   a) Storage
   b) Sharing of data
   c) Manage your calendar, to-do lists,
   d) Document editing tool

4. Exploring Microsoft cloud

5. Exploring Amazon cloud

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

PGCS-208: Lab Course - V

Course Outcomes:

At the end of the course, a student shall be able to:

- implement core Java programs to solve simple problems .
- implement Client and Server end Java programs.
- develop software in the Java programming language
- develop GUI based java Windows application

Total Credits: 02

Course content

1. Write java program to create a user defined Exception class known as Pay Out Of Bounds Exception. Organization does not offer basic salary less than 8000. If entered salary is less than 8000 then program should create an Exception of Type Pay Out Of Bounds Exception. Program should calculate gross salary by considering salary parameters such as DA, HRA, CA, TA, Professional tax, TDS, PF. etc
2. Write a Java program to identify whether inputted data is byte/short/int/long/float/double/String/char type. (Use Exception Handling)
3. Write java program to draw the house on an applet.
4. Write java program to create a registration form using AWT.
5. Write java program to create an advertisement banner on an applet using multithreading.
6. Write a Java program to demonstrate the use of AWT components namely buttons,labels, text boxes, lists/combos, menus with event handling.
7. Write a java program to store personal telephone directory in such a way that when user hits a character, the names which starts with the character and telephone numbers should appear.
8. Write a program to create a window with four text fields for the name, street, city and pincode with suitable labels. Also windows contains a button MyInfo. When the user types the name, his street, city and pincode and then clicks the button, the types details must appear in Arial Font with Size 32, Italics.
9. Write a Java program to implement Swing components namely Buttons, ,JLabels, Checkboxes, Radio Buttons, JScrollPane, JList, JComboBox, Trees, Tables Scroll pane Menus and Toolbars to design interactive GUI.
10. Write java program implementing client/server socket programming.
11. Write java programs for :-
    - Manipulate files, directories and file system
    - Create data-centric applications using JDBC
    - Process strings using regular expressions

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

PGCS-209: Lab Course - VI

Course outcomes:

At the end of the course, a student shall be able to:

- use basic concepts for building various applications in data mining field.
- understand design experiments using different data mining procedures.
- develop skills of analyzing test results of given experiments.

Total Credits: 02

Course content

Any 16 assignments on the following topics

1. Database generation and analysis using spreadsheets.
2. Data Preprocessing using WEKA: Applying filter –
   a. weka>filter>
   b. weka>filter>unsupervised>instance>
3. Explore the 'select attribute' as follows
   weka>attributeSelection>
3. Association mining
   weka>associations> , Apriori, FPGrowth
4. Classification Technique
   weka>classifiers> NaïveBayes , weka>classifiers>lazy> : IBk
   weka>classifiers>trees , Random Tree , J48
5. Clustering Technique
   weka>clusters> , SimpleKMeans , FarthestFirst algorithm, hierarchicalCluster
6. Regression analysis Technique
7. Introduction to R model
8. Comparison of Data Mining Tools
9. Case study for comparing all classification, clustering techniques for measuring errors and performance.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-II

PGAEC11: Soft Skills

Course Outcomes

At the end of this course, a student shall be able to:

- communicate with others effectively
- exhibit qualities of leadership
- take responsibility to undertake a work and complete it.
- be aware of their own weaknesses
- work in groups either as members or leaders
- think critically or laterally and solve problems
- be flexible to the needs of others
- negotiate with others to solve problems (conflict resolution)
- cope with pressure and yet produce results

Total Credits: 02 **Total Lectures: 30**

Course Contents

1. Soft Skills: Meaning and definition; Employability skills, Life skills, Corporate skills.
2. Developing positive attitude: Ethics, Values, Manners & Etiquettes
3. SWOT analysis and Career Planning: Strengths, Weaknesses, Opportunities and Threats
4. Curriculum Vitae, Resume, Bio-Data: Types of personal and educational information, preparing CV according to the job requirements
5. Interview Techniques: Types of interviews, Personal appearance, basic research, Confidence, knowledge preparation, interview style
6. Time Management: Competency-building skills; identifying use of time, management of on-time task, identifying reasons for poor time management, and taking corrective action; overall time-planning; and learning where to go for information or guidance
7. Stress Management: coping with stress and anger; recognizing/understanding others' point of view; problem solving; peer negotiation and resistance; conflict management; active listening and effective communication; acceptance and/or tolerance of diversity groups; and telephone skills
8. Confidence building and Personal Integration:
9. Teamwork and Leadership: Group discussion, effective communication skills, Group behaviour, helping nature
10. Developing work culture: sincerity, negotiation techniques, summarizing information; summarizing information about specific points; organizing studies

* * *

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS-301- Artificial Intelligence

Course outcomes:

At the end of this course, a student shall be able to:

- apply the basic principles, models, and algorithms of AI to recognize, model
- demonstrate knowledge of the building blocks of AI as presented in terms of intelligent agents.
- solve problems in the analysis and design of information systems
- analyze the structures and algorithms of a selection of techniques related to searching, reasoning, machine learning, and language processing.

Total Credits: 04 **Total lectures: 60**

Course content

1. Artificial Intelligence: **(5L)**

An overview, Intelligent Systems: Evolution of the concept.

2.Intelligent Agents : **(5L)**

how agent should act, Structure of intelligent agents, Environments

3. Problem solving **(5L)**

solving problems by searching, informed search methods, Game playing

4.Knowledge and Reasoning **(5L)**

A knowledge based agent, Representation, Reasoning Logic, Proportional logic, First order logic: Syntax and Semantics, Extensions and Notational variation ,Using first order logic

5.Building a Knowledge Base **(5L)**

Properties of good and bad Knowledge base, Knowledge Engineering, general ontology

6. Interfacing first Order Logic: **(5L)**

Interface rules involving quantifiers, An example proof, Forward and backward chaining, Completeness

7.Acting logically **(5L)**

Planning, Practical planning: Practical Planners, Hierarchical decomposition, Conditional planning

8. Uncertain Knowledge and Reasoning: **(5L)**

Uncertainty, Representing knowledge of uncertain domain, the semantics of belief networks, Inference in belief networks

9.Learning: **(5L)**

Learning from observations: General model of learning agents, Inductive learning, learning decision tree, Learning in neural and belief networks: introduction of neural networks, Perceptrons, Multilayer feed forward networks, Application of ANN, Reinforcement learning: Passive learning in a known environment, Generalization in reinforcement learning, Genetic algorithms

10.Agents that Communicate: **(5L)**

Communication as action, Types of communicating agents, A formal grammar for a subset of English

11.Expert system: **(5L)**

Introduction of expert system, Representing and using domain knowledge, Expert system shells, Explanation, Knowledge acquisition

12.Application: **(5L)**

Natural language processing, perception, Robotics

References Books:

1. Struart Russeland Peter Norvig, .Artificial intelligence: A modern Approach.
2. George F. Luger, . Artificial intelligence: Structures and Strategies for Complex Problem Solving., Person Education
3. Nils J. Nillson, .Artificial Intelligence : A New Synthesis., Harcourt Asia page 36 of 38
4. Elaine Rich and Kevin Knight,. Artificial Intelligence.,TMH
5. Patrick Winston,. Artificial Intelligence., Pearson Education
6. Ivan Brakto,.Prolog Programming for Artificial Intelligence.,Pearson Education
7. Efraim Turban Jay E.Aronson, .Decision Support System and Intelligent Systems.
8. Ed. M. Sasikumar and Others, . Artificial Intelligence :Theory and Practice.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS-302- Mobile Technologies

Course outcomes:

At the end of this course, a student shall be able to:

- get familiarize with the buzz words and technology of mobile communication
- understand SMS, MMS and GSM architecture
- apply mobile IP and mobile TCP
- understand the issues relating to wireless applications

Total Credits: 04 **Total lectures: 60**

Course content

1. Introduction to Mobile Computing: **(4L)**

Introduction and need for Mobile computing, Mobility and portability, Mobile and Wireless devices, Applications, Brief History of wireless communication

2 Wireless Transmission: (6L)

General Concepts of multiplexing and modulation, Spread Spectrum, Cellular Systems, Cellular Phone Array, Mobile Phone Technologies (1G, 2G, 2.5G, 3G)

3.Medium Access Control Layer: (6L)

Why specialized MAC?, hidden and exposed terminals, near and far terminals, General Concepts and comparison of SDMA, FDMA, TDMA ,CDMA

4. Global System for Mobile Communication: (10L)

Mobile Services (Bearer, Tele-and-supplementary services), System Architecture, Radio subsystem, Network and switching subsystem, Operation subsystem, Protocols, Localization and calling, Handover, Value Added Services, SMS, Architecture, Mobile Originated and Mobile Terminated, procedures, Cell Broadcast Service, Architecture, Message Transfer Procedure, MMS, Architecture, Protocol framework, Message Transfer, Procedure, Location Services, Logical Reference Model, Control procedures, Network Architecture, determination of Location Information, Location based services, GPRS

5.Mobile IP : (10L)

Goals, assumptions and requirements, Entities and terminologies, Agent Discovery, Registration, Tunneling and encapsulation, Reverse Tunneling, IPv6, IP micro-mobility support – Cellular IP, Hawaii, Hierarchical, mobile IPv6, Mobile Routing Destination sequence distance Vector, Dynamic Source Routing, Alternative Matrix, Adhoc Routing Protocols -Flat, Hierarchical, Geographic-position-assisted

6 Mobile TCP: (10L)

Traditional TCP, Congestion Control, Slow start, Fast retransmit / Fast recovery, Implications on mobility, Classical TCP improvements, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit / Fast recovery, Transmission / Timeout freezing, Selective Retransmission, Transaction oriented TCP, TCP over 2.5/3G wireless networks

7.Wireless Application Protocol : **(4L)**

Architecture, Wireless datagram protocol, Wireless transport layer security, Wireless transaction protocol, Wireless session protocol, Wireless application environment , WML, WML Scripts, Push Architecture, Push – Pull Services

8. Java for Wireless Devices: **(2L)**
Setting up the development environment, Basic Data types, Libraries (CLDC, MIDP)

9. **UI Controls:** **(2L)**
Displayable and Display Image, Events and Event Handling, List and choice, Text box, Alerts

10. Persistent Storage: **(2L)**
Record Stores, Records, Record Enumeration

11. **Network MIDlets:** **(2L)**
The Connection Framework, Connection Interface, Making a connection using HTTP, Using datagram connection

12. Wireless Messaging: **(2L)**
Architecture for Messaging application, Messaging API, Types of applications, Pro's and con's of messaging

Reference Books:

1. Mobile Communication By Jochen Schiller
2. Pervasive Computing Technology And Architecture Of Mobile Internet Applications by JOCHEN BURKHARDT and HORST HENN, PEARSON INDIA
3. Mobile Computing by Rajkamal, Oxford University Press
4. Mobile Computing by TALUKDER and ASOKE K, McGraw Hill

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS-303: .Net Technologies

Course outcomes:

At the end of this course, a student shall be able to:

- understand the development and deployment cycles of enterprise applications.
- develop ASP.NET Web Services, secure web services, and .NET remote applications.
- Develop web applications using a combination of client-side and server side technologies
- experiment with the Windows Form.

Total Credits: 04 **Total lectures: 60**

Course content

1. Introduction to .Net Technology: (6L)

Framework common Language Runtime, Common Language Specification, Intermediate Language Code, Just-In-Time Compiler, Assemblies, Manifest, Metadata ,Global assembly Cache, VB.NET, Creating Applications, Building, Running Projects, Data , Operator, Loops, Procedures, Functions.

2. Windows Forms: (6L)

Text Box, Buttons, Labels, Checks Boxes, radio Buttons, List Boxes, Combo Boxes, Picture Boxes, Scrollbars, Timer, Menus, Built-in Dialogs, Image List, Toolbars, Status Bar and Progress bars, Event and Delegates, Tracing, Debugging

3. Object Oriented Programming in VB.NET: (6L)

Class and Objects Properties, methods and events, Contractor and Destructor, Method overloading, Inheritance, Access modifiers, Public, Private, Protected, Friend, Overriding and shadowing, Interfaces, Polymorphism, Error Handling, Private and Shared Classes

4. File Handling: (6L)

Stream Writer, Stream Reader, MSDN, Binary Reader, Binary Writer Classes MSDN, File and Directory Classes

5. ADO.Net Connections: (6L)

Data adapters, and Datasets ,Data binding with controls, Navigating data source ,Data from wizard, Data reader, Connection objects, command Objects, Data Adapters, Dataset Class, Data Readers, Non-Queries, Query-Parameters, Scalar-Queries, Calling stored-Procedures, Data-Relations, Editing rows ,Filtering and sorting, XML (Reading, Writing)

6. Web application: (8L)

Introduction to Web form, page directive, all validation controls, Page redirection

7. State management: (2L)

Application state, Session State, View state, Web controls, Tracing web applications

8. Database: (6L)

Data grid control in web applications

9. Web services: (4L)

Concept of web services, MSDN, Create a small web services

10. Deployment: (2L)

Deploying applications using wizard MSDN

11. Security introduction: **(2L)**
Securing a Microsoft Applications, MSDN

12. C#: **(4L)**
Introduction to C#, Windows Forms with C#, Difference between C# and VB.Net MSDN Textbox, Label, Command button

13. Object Oriented Programming in C#.Net: **(2L)**
Class and Objects, Properties, Methods and events, Contractors and Destructor, Method overloading, Inheritance, Interface

Reference Books:

1. MCAD / MCSD:Developing Windows-Based Applications With Microsoft Visual Basic.Net and Visual C# .Net By Mattjew Stoecker
2. MCAD / MCSD:Developing Web-Based Applications With Microsoft Visual Basic.Net and Visual C# .Net By Jeff Webb
3. Programming Microsoft VB.NET Microsoft Press By Belena
4. Programming Microsoft VB.NET ASP.NET By Dino Esposito

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS – MIII: MINOR PROJECT-III

Learning outcomes:

At the end of the course, a student shall be able to:

- apply knowledge and techniques learnt in theoretical classes for developing the s/w for real problems.
- Get an insight into the working of the real organizations/companies.
- gain deeper understanding in specific functional areas.
- exploring career opportunities in their areas of interest.

Total Credits: 04

Course content

The objective of this minor project is to gear up student for preparation of final project training in Semester-IV.

Student will select individually Commercial or Technical project based on Technologies learnt in Semester I. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation. The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

- The Project can be platform, Language and technology independent.
- Project will be evaluated by project guide.
- Assessment will be done weekly in the respective batch.
- Evaluation will be on the basis of weekly progress of project work, progress report, oral, results and documentation and demonstration.
- You should fill your status of the project work on the progress report and get the Signature of project guide regularly.
- Progress report should sharply focus how much time you have spent on specific task. (The format of progress report is given as follow.)
- You should keep all signed progress report.
- Project will not be accepted if progress report is not submitted and all responsibility remains with student.
- Students should prepare design document using SE/UML techniques depends on your project.

About project Report: -

The report should be typed on A4 size, executive bond paper for the final submission. The report should be in the good quality Rexene bound. We suggest, using one-and-half spaced printing, Times New Roman 12 font sizes for the normal text, 14-16 font sizes for headings & page titles. Number of copies: For one project you should prepare 2 copies of the project report. One for yourself, one for college (College copy can be in CD).

Evaluation for internal 40 marks

Description	*Marks*
UML /ERD/DFD diagrams	10
Technology and design base first demo	10
Project technology based two assignments	10
Second Demo	10

Evaluation for external 60 marks

Description	Marks
Demo	10
Report	10
Presentation	20
Viva	20

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

CORE ELECTIVE-III

PGCS-304: Software Architecture

Course outcomes:

At the end of this course, a student shall be able to:

- design and motivate software architecture for large-scale software systems
- implement major software architectural styles, design patterns, and frameworks
- make use of various software architectural styles to design modern systems
- apply software paradigms and software architecture

Total Credits: 04 **Total lectures: 60**

Course content

Pre-requisites:

Some familiarity with modern software engineering concepts and experience in designing and developing software systems.

SYLLABUS:

1. **Introduction :** **(8L)**

Basic Concepts, Introduction and need of Software Architecture , Software Architecture Design Plan,Software Architecture as an abstraction , Software Architecture terminology, Different Engineering concerns addressed by different views.

2. **Designing, Describing and using Software Architecture:** **(8L)**

System Overview,Product Features, System Interactions, The future of IS2000

3. **Global Analysis:** **(6L)**

Overview of Global Analysis Activities, Analyze factors, Develop strategies, Analyze Organisational , Technological, Product Factors, Global Analysis Summary

4. **Conceptual Architecture view:** **(10L)**

Design Activities for the conceptual, Architecture view, Design of Conceptual view for IS2000, Summary of Conceptual Architecture view, Traceability. Uses for the Conceptual Architecture view.

5. **Module Architecture View:** **(8L)**

Design Activities for the Module Architecture view, Design of Module view for IS2000. Uses for the Module Architecture view.

6. **Execution Architecture View:** **(6L)**

Design Activities for the Execution Architecture View, Design of Execution View for IS2000.Design summary for IS2000 Execution View, Summary of Execution Architecture View And Traceability. Uses for the Execution Architecture View.

7. **Code Architecture View :** **(6L)**

Design Activities for the Code Architecture View, Design of code Architecture View for the IS2000.Summary of code Architecture, View and Traceability, Uses for the code Architecture View

8. **Software Architecture Best Practice:** **(4L)**

Global Analysis, Conceptual Architecture View, Module Architecture View, Execution Architecture View, Code Architecture View, Software Architecture Uses

9. **Role of software Architecture:** **(4L)**

Creating a Vision, The Architect as a Key Technical Consultant, The Architect makes decisions, The Architect Coaches, The Architect Coordinates, The Architect Implements

Reference Books:

1. Applied software architecture, C. Hofmeister, R. Nord, D. Soni, Addison Wesley, (2000)
2. Software paradigms, S.H. Kaisler, John Wiley & sons, (2005).
3. Design Patterns: Elements of Reusable Object - Oriented software, E. Gamma, R. Helm, R. Johnson, J. Vlissides, Addison – Wesley (1995).

* * * * * *

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

CORE ELECTIVE-III PGCS-305: Software Testing

Course Outcomes:

At the end of this course, a student shall be able to:

- apply the principles of system and component testing.
- gain knowledge about strategies for generating system test cases.
- analyze understand the essential characteristics of tool used for test automation.
- identify differences between validation testing and defect testing.

Total Credits: 04 **Total lectures: 60**

Course content

Pre - requisites:

Some familiarity with modern software engineering concepts and experience in designing and developing software systems.

1: Big Picture: Introduction: (6L)

Famous case studies. What is a bug? Software failure terminologies, Why do bugs occur? Cost of bugs in software, Categories, Consequences, Testing Fundamentals. Test Axioms. Six Essentials of testing, Tester:? Competencies. Assessing Quality of the organization.

2: The S/W Development Process: (6L)

Realities of S/W testing, Life Cycle Models, Building a s/w testing process. Realities of S/W testing, Types of testing. Testing terms and definitions. Developing a test plan, Requirements verification checklist. Case study, Workbench, Do Procedures, Check Procedures.

3: Testing Methods: (6L)

Verification Testing, Validation Testing, What are test cases? Creating test cases. Test case planning overview, Reducing the number of test cases.

4: Eleven Step Testing Process, Testing Process Overview: (6L)

Clean Sheet Approach, Eleven Steps, Issues in requirements testing. Black-box, White-box, Static and Dynamic Testing.

5: Examining code, Issues in Code Testing. Static methods, Case Study (10L)

Testing Software with X-Ray glasses, Dynamic testing, Coverage Types. Validation, Validation Testing, Controlling validation Costs, Alpha & Beta Testing, Software Testing tools, Types continued, Need for Tools, Types.

6: Test Documentation & Reporting (10L)

Bug Reporting, Bug Tracking Systems, Measurement Types. Metrics, Recommendations. Types of testing: More categories, Flow Testing, Path testing, Syntax Testing, Transition Testing, Compatibility Testing, Usability Testing, Configuration Testing, Web Site Testing, Foreign Language Testing.

7: Software Quality Assurance (10L)

CMM, Career: Software Tester, Hands-On Experience. Training, Professional Organizations, Managing Test Technology.

8: Current Practices & Trends **(6L)**

What is new? Developer to Tester Ratio? Benchmarks, Test Security. Testing off-shelf s/w, Identification and precautions.

Reference Books:

1. Software Testing: Ron Patton
2. Effective Methods for Software Testing: William E Perry, Wiley
3. Software Testing in the Real World: Edward Kit
4. Introducing Software Testing: Louise Tamres
5. Software testing Techniques: Boris Beizer
6. Effective Methods for Software Testing: William Perry, Wiley
7. Software Testing: Renu Rajani

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

CORE ELECTIVE-III PGCS-306: Advanced Operating System

Course Outcomes:

At the end of this course, a student shall be able to:

- manage various OS components like System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems.
- and implement the policies for scheduling, deadlocks, memory management, synchronization, system calls, and file systems
- analyze the various device and resource management techniques for timesharing and distributed systems.
- analyze simple concurrent programs using transactional memory and message passing and to understand the trade-offs and implementation decisions.

Total Credits: 04 **Total lectures: 60**

Course content

1. Overview of Operating systems: (4L)

Objectives and Functions, Essentials of modern operating systems, Overview of UNIX, LINUX and Windows

2. **Process Management and Scheduling: (6L)**

Process and its states, Process control, Scheduling algorithms, Multi-processor and real time scheduling, UNIX Process management & scheduling, Windows Scheduling

3. **Threads and Symmetric Multi-processing (4L)**

Processes and Threads, SMP, Implementation of above in Popular Operating Systems

4. **Concurrency Control (6L)**

Mutual exclusion and synchronization, Deadlocks and Starvation**,** Mutual Exclusion & concurrency, Semaphores, Monitors, message Passing, Readers/Writers, Problem

5. **Memory Management and Virtual Memory (12L)**

Memory Management, Partitioning, Paging and Segmentation, Virtual memory and associated algorithms, Implementation of above in Windows and UNIX

6. **I/O management and File management (12L)**

I/O devices, I/O functions, I/O buffering, Disk Scheduling, RAID levels & Disk cache ,Virtual File Systems, UNIX and Windows support for I/O, File Organisation, directories, sharing, Storage management and Record Blocking, UNIX and Windows File management

7. **Networking and Security: (8L)**

Protocol architecture, TCP/IP, Sockets, Network File System, Security Threats, protection, Intruders, malicious software, trusted systems

8. **Distributed processing, client/servers and Clusters (8L)**

Client/server computing, message passing, RPC Clusters

References Books:

1. Operating Systems – Internals and Design principles by William Stallings (Fifth Edition, Prentice Hall India – 2005)

2. The MINIX Book: Operating System – Design and Implementation by Andrew S Tanenbaum and Albert S Woodhull (Third Edition, Prentice Hall India – 2006)
3. UNIX Systems for Modern Architectures by Curt Schimmel, Addison Wesley
4. Applied Operating Systems Concepts, Seventh Edition by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, John Wiley and Sons, 2004

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS - 307: Lab Course -VII

Course Outcomes:

At the end of this course, a student shall be able to:

- apply software testing knowledge and engineering methods
- apply a software test process for a software testing project
- develop ability to identify the needs of software testing automation
- define and develop the test tool to support test automation

Total Credits: 02

Course content

1. Practical Title
• Problem Statement,
• Process Model

2. Requirement Analysis
• Creating a Data Flow
• Data Dictionary,
• Use Cases

3. Project Management
• Computing FP
• Effort
• Schedule, Risk Table, Timeline chart

4. Design Engineering
• Architectural Design
• Data Design, Component Level Design

5. Testing
• Basis Path Testing

6. Case study on software cost estimation

* * * * * *

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS - 308: Lab Course- VIII

Course Outcomes:

At the end of this course, a student shall be able to:

- develop applications using .net framework for C# programming languages.
- perform connectivity application using ADO.Net.
- perform web services.

Total Credits: 02

Course content

- Attain a detailed working knowledge of C-Sharp implicit types, object initializers, delegates, anonymous types & methods, extension methods, lambda expression, LINQ and many more.
- Acquire a working knowledge of creating and rich internet Web application using the .NET Framework 4.5 and Visual Studio 2012.
- Attain a detailed knowledge of the building blocks of Web application, including C-Sharp, ASP.NET, ADO.NET, Web Services and Ajax.
- Understand the ASP.NET programming model issues, such as state, caching and data handling management.
- Configure and deploy a Microsoft ASP.NET Web application.Our dot net training in Chennai will help you gain a strong knowledge in all Dot Net Web application concepts
- Produce ASP.NET programs for the web using solid multilayer architectures.
- Learn how to implement web applications using web forms, including programs that interact with databases.
- Work with data from multi sources like objects, XML, databases using Language Integrated Query (LINQ) and the Entity Framework (EF).
- Develop a clean, maintainable code base using the Model View Controller (MVC 4) architecture.
- Create flexible views for user interaction with view helpers.
- Learn to manage data access tasks by using LINQ.
- Learn to create a Microsoft ASP.NET AJAX application.
- Learn to consume and Windows Communication Foundation (WCF) services.
- Learn to manage data by using ASP.NET Dynamic Data.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGCS - 309: Lab Course - IX

Course Outcomes:

At the end of this course, a student shall be able to:

- perform various Linux and Unix command for managing operating system as well as user.
- to understand the basic commands of Linux operating system and can write shell scripts
- create file systems and directories and operate them.
- create shared memory segments, pipes, message queues and can exercise interprocess communication

Total Credits: 02

Course content

Note: Following exercises can be performed using Linux or Unix

1. Usage of following commands: ls, pwd, tty, cat, who, who am I, rm, mkdir, rmdir, touch, cd.
2. Usage of following commands: cal, cat (append), cat (concatenate), mv, cp, man, date.
3. Usage of following commands: chmod, grep, tput (clear, highlight), bc.
4. Write a shell script to check if the number entered at the command line is prime or not.
5. Write a shell script to modify "cal" command to display calendars of the specified months.
6. Write a shell script to modify "cal" command to display calendars of the specified range of months.
7. Write a shell script to accept a login name. If not a valid login name display message – "Entered login name is invalid".
8. Write a shell script to display date in the mm/dd/yy format.
9. Write a shell script to display on the screen sorted output of "who" command along with the total number of users.
10. Write a shell script to display the multiplication table any number,
11. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
12. Write a shell script to find the sum of digits of a given number.
13. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
14. Write a shell script to find the LCD (least common divisor) of two numbers.
15. Write a shell script to perform the tasks of basic calculator.
16. Write a shell script to find the power of a given number.

*** * * * * ***

M.Sc. (Computer Science) (CBCS 2018 Course) Semester-III

PGSEC31: **Android Programming**

Course Outcomes:

At the end of this course, a student shall be able to:

- understand Android platform and its architecture.
- learn about mobile devices types and different modern mobile operating systems.
- learn activity creation and Android User Interface designing.
- learn basics of Intent, Broadcast and Internet services.
- understand different approaches to structuring the applications
- understand application life cycle

Total Credits: 02 **Total lectures: 30**

Course content

1.Introduction: **(4L)**

History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.

2.Overview of object oriented programming using Java: **(6L)**

OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

3.Development Tools: **(8L)**

Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project – Hello Word, run on emulator, Deploy it on USB-connected Android device.

4.User Interface Architecture: **(4L)**

Application context, intents, Activity life cycle, multiple screen sizes.

5.User Interface Design: **(4L)**

Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners (Combo boxes), Images, Menu, and Dialog.

6.Database: **(4L)**

Understanding of SQLite database, connecting with the database.

Reference Books::

1. Android application development for java programmers. By James C. Sheusi. Publisher: Cengage Learning, 2013.

Online Reading / Supporting Material:

1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://docs.oracle.com/javase/tutorial/index.htm (Available in the form of free downloadable ebooks also).
5. http://developer.android.com/guide/components/activities.html
6. http://developer.android.com/guide/components/fundamentals.html

7. http://developer.android.com/guide/components/intents-filters.html.
8. http://developer.android.com/training/multiscreen/screensizes.html
9. http://developer.android.com/guide/topics/ui/controls.html
10. http://developer.android.com/guide/topics/ui/declaring-layout.html
11. http://developer.android.com/training/basics/data-storage/databases.html

Software Lab Based on Android Programming:

1. Create "Hello World" application. That will display "Hello World" in the middle of the screen in the emulator. Also display "Hello World" in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.
6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

*** * * * * ***

M.Sc. (COMPUTER SCIENCE) (CBCS 2018 Course) Semester- IV

PGCS – 401: Internship

Course Outcomes:

At the end of the course, a student shall be able to:

- survey and study of published literature on the topic selected
- Work out a preliminary approach to the Problem relating to the topic
- Conduct preliminary Analysis/ Modelling/ Simulation/ Experiment/ Design/ Feasibility
- preparing a written report on the study conducted for presentation to the Department
- final Seminar, as presentation of the actual project undertaken before the exam committee

Total Credits: 18

Course content

Fourth semester is internship for 200 marks. Project work may be done individually or in groups in case of bigger projects. However if project is done in groups, each student must be given a responsibility for a distinct module and care should be taken to see the progress of individual modules is independent of others.

Students should take guidance from an internal guide and prepare a Project Report on "Project Work" in 2 copies to be submitted in the department by first week of March. The Project Report should contain an Introduction to Project, which should clearly explain the project scope in detail. Also, Data Dictionary, DFDs, ERDs, File designs and a list of output reports should be included.

Project work can be carried out in the Institute or outside with prior permission of the Institute. Project viva-voce by the University panel comprising of internal examiner, external examiner and one industry expert will be conducted in the month of March-April.

General Instructions Regarding Preparation of Project Report for M.Sc. (Comp. Sci.) SEM- IV

TYPING

1. The typing shall be standard 12 pts in double spaced using black ink only
2. Margins must be Left 2 inches Right 1.5 inches Top 2 inches Bottom 1.5 inches
3. Paper A4 size Bond Paper

COPIES

Two hard-bound copies
(Black Rexine with Golden Embossing as per format displayed herewith)
One original and one clean Xerox Copy.

Project Evaluation Phases:

Recommended Phase	Description	Timeline
1	SRS Document	3nd Week
2	Design document	7th Week
3	Executable/User Interface	12th Week
4	Test plan and Documentation	16th Week
5	Project Viva/Presentation	20th Week

* * *

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY), PUNE

Faculty of Science
M.Sc - Computer Science
Old Syllabus

"Social Transformation Through Dinamic Education"

BHARATI VIDYAPEETH (DEEMED TO BE UNIVERSITY)

YASHWANTRAO MOHITE COLLEGE OF ARTS, SCIENCE AND COMMERCE, PUNE 411038

Accredited with 'A+' Grade (2017) by NAAC
'A' Grade University Status by MHRD, Govt. of India
Accredited (2004) & Reaccredited (2011) with 'A' Grade by NAAC

MASTER OF COMPUTER SCIENCE (M.Sc. Computer Science) PROGRAMME

CBCS 2012 COURSE STRUCTURE

Under the Faculty of Science

TO BE IMPLEMENTED FROM ACADEMIC YEAR 2012-13

BHARATI VIDYAPEETH
DEEMED UNIVERSITY, PUNE (INDIA)

Master of Computer Science M.Sc.(Computer Science)
(Choice Based Credit and Grade System)
Under: Faculty of Science
(To be implemented from 2012)

The Master of Computer Science, M.Sc (Computer Science) Program is a full time 100 Credits program offered by Bharati Vidyapeeth Deemed University (BVDU), Pune The expectations and requirements of the Software Industry are visualized while designing the M.Sc(Computer Science).

1. Objectives : M.Sc(Computer Science) Course:

The main objective of the M.Sc. Computer Science course is to provide the students a clear understanding of the basic concepts and principles of Computer Science as a discipline and the rich and specialized skill sets required to handle the computing systems in an applied branch of knowledge that represent a realistic world. All aspects of knowledge acquisition , storage , processing , presentation and transition will be covered in the course. Provision for advanced knowledge in Computer Science, exposure to the practical and theoretical concepts of computing, current and emerging trends in technology in the context of networking environment are the major attractions of the programme.

The course covers almost all the core subjects in Computer Science like Design and Analysis of Algorithm, Network Programming, OOP, Software Engineering, Database Concepts etc, in addition to a few elective subjects. The Case Studies in all the three semesters intensively focus on the systematic design and implementation of various technological applications through which the students attain the skills in system development and team spirit. As part of the course, the students are advised to take live projects of 100 to 120 days duration from established institutions for implementation during their project term. In addition, 'communication skills', 'life skills' and 'research skills' which are necessary for career growth and for leading quality life are also imparted.

Learning Outcomes from the M.Sc (Computer Science)

At the end of the course the student should be able to

(a) Analyze problems and design effective and efficient software solutions
(b) Develop software under latest Application Development Environments.
(c) Learn new technologies with efficient implementation
(d) Read, write, and contribute to technical literature
(e) Work in team
(f) Socially committed IT professionals

2. Eligibility for Admission to the Course:

A student shall be eligible for admission to the First Year M.Sc. (Computer Science) degree course who has completed B.Sc.(Computer Science) / B.Sc(Computer application) graduation from any recognized university satisfying the following

conditions. The candidate should have secured at least 50% (45% for SC/ST) in aggregate at graduate level university examination.

3. Intake Capacity:

The intake capacity of the course will be 50 seats every year.

4. Course Structure

The M.Sc. (Computer Science) course will be of minimum four semesters and with a minimum of 100 credits. The medium of instruction and examination will be only English. The credit allotment for M.Sc.(Computer Science) course : Semester I (28 Credits), Semester II (28 Credits), Semester III (28 Credits), and Semester IV (16 Credits). In each Semester, there will be four papers(three core compulsory and one core elective) of 100 marks each, Laboratory course and minor project for each Semester of 100 marks each, out of which 40 marks will be for internal assessment and 60 marks for university examination. Fourth semester is internship for 200 marks. Thus M.Sc.(Computer Science) degree examination, four semesters shall be 2000 marks and of minimum 100 credits altogether . The following shall be the course structure:

SEMESTER-WISE COURSE INFORMATION

SEMESTER I

Semester	Subject Type	Code	Title of the paper	Credit Value	Lec.	T	Lab.	Weightage for EoTE/IA	EoTM
Semester I	Core: Compulsory	PGCS-101	Algorithm Design Patterns	5	4	1	-	0.6/0.4	Univ.
		PGCS-102	Paradigm of programming Languages	5	4	1	-	0.6/0.4	Univ.
		PGCS-103	Advanced Data Structures	5	4	1	-	0.6/0.4	Univ.
	Core: Elective	**Any one from the following:**							
		PGCS-104	Parallel Processing	5	4	1	-	0.6/0.4	Univ.
		PGCS-105	Theory of Automata	5	4	1	-	0.6/0.4	Univ.
		PGCS-106	Digital Image Processing	5	4	1	-	0.6/0.4	Univ.
		PGCS-107	Lab Course –I	4	-	-	3	0.6/0.4	Univ.
		PGCS-108	Minor Project –I	4			3	0.6/0.4	Univ.
	TOTAL			28	18	04	06		
		To earn extra credits, a student may opt for any number of the following subjects:							
	Non-Core: Optional	PGGEN11	English For Specific Purposes	2	2	-	-	0.6/0.4	Univ.
		PGGEN12	Basics of Information Technology	2	2	-	-	0.6/0.4	Univ.
		PGGEN13	Bioinformatics	2	2	-	-	0.6/0.4	Univ.

SEMESTER II

Semester	Subject Type	Code	Title of the paper	Credit Value	Lec.	T	Lab.	Weightage for EoTE/IA	EoTM
Semester II	Core: Compulsory	PGCS-201	Software Architecture	5	4	1	-	0.6/0.4	Univ.
		PGCS-202	Network Security	5	4	1	-	0.6/0.4	Univ.
		PGCS-203	Java Application Programming	5	4	1	-	0.6/0.4	Univ.
	Core: Elective	**Any one from the following:**							
		PGCS-204	Software Testing	5	4	1	-	0.6/0.4	Univ.
		PGCS-205	Embedded Computing	5	4	1	-	0.6/0.4	Univ.
		PGCS-206	Software Project Management	5	4	1	-	0.6/0.4	Univ.
		PGCS-207	Lab Course –II	4	-	-	3	0.6/0.4	Univ.
		PGCS-208	Minor Project -II	4			3	0.6/0.4	Univ.
			Total	28	18	04	06		
		To earn extra credits, a student may opt for any number of the following subjects:							
	Non-Core: Optional	PGGEN21	Consumer Protection	2	2	-	-	0.6/0.4	Univ.
		PGGEN22	Scientific Writing	2	2	-	-	0.6/0.4	Univ.
		PGGEN23	Cyber Security	2	2	-	-	0.6/0.4	Univ.

SEMESTER III

Semester	Subject Type	Code	Title of the paper	Credit Value	Lec.	T	Lab.	Weightage for EoTE/IA	EoTM
Semester III	Core: Compulsory	PGCS-301	Artificial Intelligence	5	4	1	-	0.6/0.4	Univ.
		PGCS-302	Mobile Technologies	5	4	1	-	0.6/0.4	Univ.
		PGCS-303	.Net Technologies	5	4	1	-	0.6/0.4	Univ.
	Core: Elective	**Any one from the following:**							
		PGCS-304	Soft Computing And Data Mining	5	4	1	-	0.6/0.4	Univ.
		PGCS-305	Advanced Database Management Systems	5	4	1	-	0.6/0.4	Univ.
		PGCS-306	Advanced Operating System	5	4	1	-	0.6/0.4	Univ.
		PGCS-307	Lab Course –III	4	-	-	3	0.6/0.4	Univ.
		PGCS-308	Minor Project –III	4			3	0.6/0.4	Univ.
			Total	28	18	04	06		
		To earn extra credits, a student may opt for any number of the following subjects:							
	Non-Core: Optional	PGGEN31	Developing Soft Skills	2	2	-	-	0.6/0.4	Univ.
		PGGEN32	Clinical Biochemistry	2	2	-	-	0.6/0.4	Univ.
		PGGEN33	Income Tax Management	2	2	-	-	0.6/0.4	Univ.

SEMESTER IV

Semester	Course Number	Course Title	Credit Value	EoTM
Semester IV	PGCS-401	Internship	16	Univ.

5. Scheme of Examination:

In order to pass in a course, the student must obtain a minimum of 'D' grade at the UE and a GPI of 4.5 in aggregate of University Examination and Internal Assessment .For each paper of 100 marks , there will be internal assessment of 40 marks and the university examination of 60 marks for 3 hours duration.

The Internal Assessment (IA) of each course will be for 40 marks and conducted by the college during the term. The assessment may be based on evaluations of the following: Written tests, seminar/presentation by the student, assignment, orals or any other. Hence there must be at least two evaluations of different kind.

Each practical examination for laboratory course is of 100 marks and three hour duration. The minor projects in Semesters I ,II and III will be evaluated for 100 marks for the allotted credits by a panel consisting of one internal and one external examiner .For both laboratory course and minor project , there will be internal assessment of 40 marks and the university examination of 60 marks.

A candidate shall be permitted to proceed from the first semester upto final semester irrespective of their failure in any of the semester examinations subject to the condition that the candidate should register for all the arrear subjects of earlier semesters along with current(subsequent)semester subjects.

At the end of each semester, a cumulative grade point average (CGPA) and also Semester grade point average (SGPA) will be calculated as a weighted average of the GPI of all courses in which the student has passed till that semester.

A student who passes in all the courses will be declared to have passed the entire M.Sc (Computer Science) with the following honours.

CGPA in [4.00, 4.99] -- Pass Class

CGPA in [5.00, 5.49] -- Second Class

CGPA in [5.50, 5.99] -- Higher Second Class

CGPA in [6.00, 7.99] -- First Class

CGPA in [8.00, 10.00] – First Class with Distinction

6. STANDARD OF PASSING:

A candidate shall be declared to have passed in the paper provided he/she has secured minimum 'D' grade in the university examination and a GPI of 4.5 in aggregate of UE and IA..

7. Scheme of credits:

The M.Sc(Comp.Sci) is of 100 credits. The distribution of credits over semesters is given below.

Course Type	Credits	Sem-I		Sem-II		Sem-III		Sem-IV	Total Credits	EoTE Univ/College
		L/pw	T	L/pw	T	L/pw	T			
Core Compulsory Theory	5	4 60H	2 30H	4 60H	2 30H	4 60H	2 30H		15C 270H	University
Core Compulsory Theory	5	4 60H	2 30H	4 60H	2 30H	4 60H	2 30H		15C 270H	University
Core Compulsory Theory	5	4 60H	2 30H	4 60H	2 30H	4 60H	2 30H		15C 270H	University
Elective theory	5	4 60H	2 30H	4 60H	2 30H	4 60H	2 30H		15C 270H	University
Laboratory course	4	6 92H	-	6 92H	-	6 92H	-		12C 276H	University
Project Work	4	4 60H	-	4 60H	-	4 60H	-		12C 180 H	University
Internship		-						16C 480H	16C 480H	University
Total Credits Per Semester		28		28		28		16	100	
Total Hours Per Semester		392 H		392 H		392 H		480 H	1656H	

1 Credit for theory : 15 Hrs
1 Credit for tutorial : 30 Hrs
1 Credit for Laboratory course : 23 Hrs
1 Credit for Project work : 15 Hrs

8. Extra Credits

Regular students can also opt for extra credits if the departmental committee agrees .A candidate must communicate which extra credits she/he is going to attempt at the commencement of the semester. A student can enroll for extra credits over and above the total 100 credits prescribed for the course by attending and appearing for the examination of the

opted extra credits. However in such case grades of the extra credit will NOT be counted for arriving at GPA. The extra credit course can be selected from within the institute or any other faculties of Bharati Vidyapeeth University offering courses for Faculty of Science, Faculty of Arts, Social Sciences and Commerce which will be declared at the commencement of session. Such extra credits will be mentioned separately in the grade sheet.

9. Grade point scheme

The 10-point scale would be used to convert marks out of 100 to grade and grade points according to the following table:-

Marks as Percentage	Grade	Grade Point
[75.0, 100]	O	10.0
[70.0, 74.9]	A+	9.0
[65.0,69.9]	A	8.0
[60.0,64.9]	B+	7.0
[55.0,59.9]	B	6.0
[50.0,54.9]	C+	5.5
[45.0,49.9]	C	5.0
[40.0,44.9]	D	4.5
[00.0, 39.9]	F	0.0

10. STRUCTURE OF TRANSCRIPT

At the end of each semester, student will be given a transcript showing the performance and result in each course. The transcript shows, for each course the title of the course, credit values, grade in UE, grade in IA, Grade point index, result as pass or fail. Also, the Semester grade point average and cumulative grade point average will be shown. Further, the equivalent percentage of marks corresponding to SGPA and CGPA will be shown. The formula to convert SGPA or CGPA to equivalent percentage is given by:

$$\text{Equivalent percentage marks} = \begin{cases} 10 \times \text{CGPA} & \text{if CGPA/SGPA is in } [4.00, 6.00] \\ 05 \times \text{CGPA} + 30 & \text{if CGPA/SGPA is in } [6.00, 9.00] \\ 25\text{x CGPA} - 150 & \text{if CGPA/SGPA is in } [9.00,10.00] \end{cases}$$

The above formula computes to the following table:-

Range in % of marks	Range of CGPA	Letter grade	Division
[75.0, 100]	[9.00, 10.00]	O	Distinction
[70.0, 74.9]	[8.00,8.99]	A+	
[65.0,67.9]	[7.00,7.99]	A	First class
[60.0,64.9]	[6.00,6.99]	B+	
[55.0,59.9]	[5.50,5.99]	B	Higher second class
[50.0,54.9]	[5.00,5.49]	C+	Second class
[45.0,49.9]	[4.50,4.99]	C	Pass class
[40.0,44.9]	[4.00,4.49]	D	
[00.0, 39.9]	[0.00,3.99]	F	Fail

Thus, the percentage of marks can be obtained by using the following table:

CGPA	% of Marks	CGPA	% of Marks	CGPA	% of Marks	CGPA	% of Marks
4.0	40.0	**5.5**	55.0	**7.0**	65.0	**8.5**	72.5
4.1	41.0	**5.6**	56.0	**7.1**	65.5	**8.6**	73.0
4.2	42.0	**5.7**	57.0	**7.2**	66.0	**8.7**	73.5
4.3	43.0	**5.8**	58.0	**7.3**	66.5	**8.8**	74.0
4.4	44.0	**5.9**	59.0	**7.4**	67.0	**8.9**	74.5
4.5	45.0	**6.0**	60.0	**7.5**	67.5	**9.0**	75.0
4.6	46.0	**6.1**	60.5	**7.6**	68.0	**9.1**	77.5
4.7	47.0	**6.2**	61.0	**7.7**	68.5	**9.2**	80.0
4.8	48.0	**6.3**	61.5	**7.8**	69.0	**9.3**	82.5
4.9	49.0	**6.4**	62.0	**7.9**	69.5	**9.4**	85.0
5.0	50.0	**6.5**	62.5	**8.0**	70.0	**9.5**	87.5
5.1	51.0	**6.6**	63.0	**8.1**	70.5	**9.6**	90.0
5.2	52.0	**6.7**	63.5	**8.2**	71.0	**9.7**	92.5
5.3	53.0	**6.8**	64.0	**8.3**	71.5	**9.8**	95.0
5.4	54.0	**6.9**	64.5	**8.4**	72.0	**9.9**	97.5
						10.0	100.0

11. Internship

The assessment for 16 credits in Internship during IV semester of M.Sc. (Computer science) will be as follows:

i)A student will inform the department about the joining date of the above mentioned training .

ii)The student will have to make two presentations: at the mid of the training and at the end of the training programme. The student will have to submit a compiled report which will be assessed towards course credit.

Upon completion of the Internship, the student will submit a report based on the work done. The report will be evaluated based on viva and presentation by one internal examiner ,one external examiner and industry expert. The internship assessment will for 200 marks ,out of which 80 marks will be evaluated by internal examiner and 120 marks will be evaluated by external examiner and industry expert

PGCS-101: ALGORITHM DESIGN PATTERNS

(5Credits, 4L+2T)

Objectives:

1. Emphasis on design techniques. Acknowledge of design will certainly help one to create good algorithms.
2. The students should be inhibited from taking a cookbook approach to algorithm design by assuming that each algorithm must derive from only a single technique.
3. The major goal of this organization is to emphasize the arts of synthesis & analysis of algorithms.

References :-

1) Fundamentals of computer Algorithms – Coreman
2) Algorithm Design – Kleinberg and Tardos

SYLLABUS:

1 .Introduction: (4L)

Algorithm Concept, Writing structured programs, Analyzing algorithms

2. Divide & conquer: (12L)

The General method, Binary search, Finding the maximum &minimum selection, merge sort

3. The greedy method: (10L)

The general method, Optimal storage on tapes, Knapsack problem, Job Sequencing with deadlines, Optimal merge patterns, Minimum spanning tree

4. Dynamic programming: (10L)

The general method, Multistage Graphs, All pairs shortest path, Optimal binary search tree, Flow shop scheduling

5. Basic search & traversal techniques: (12L)

The techniques, code optimization, AND/OR graphs Game tree , Backtracking . The general method, The 8-queens problems, Sun of subsets, Graph coloring, Hamilton cycles

6. Branch-and-Bound: (8L)

Strategy ,The method 0/1 knapsack problem, Traveling salesperson, LIFOBB, FIFOBB

7. NP-HARD&NP- COMPLETE problem: (4L)

Basic concepts, cook's theorem, NP-HARD graph problem, NP-HARD scheduling problem

PGCS-102: PARADIGMS OF PROGRAMMING LANGUAGES

(5Credits, 4L+2T)

Objectives:

1) Exposing the students to basic concepts, paradigms and syntax of different languages.

2) Provide an overview of the key paradigms used in developing modern programming languages.

3) Explore the implementation of each language in sufficient details to provide the programmer and understanding of the relationship between a source programmer and its execution behavior.

4) Provide sufficient formal theory to show were programming languages design fits within the general computer science, research agenda.

Prerequisite:

To understanding of structured programming, procedural programming and object oriented programming.

Reference :-

1. Programming languages design $ implementation by terrace W.Pratt, Marvin V Zelkowit
2. Programming paradigms Methology by G.P.Potdar

SYLLABUS:

1. **Introduction to Programming languages: (10 L)**
   Need for studying programming languages, A Short of programming languages, Features of a good language, Effects of environments on languages

2. **Principles of languages: (10 L)**
   Design Structure and operation of a computer Virtual computers and languages Implementations syntax, type and semantics, Context free grammars, Grammar For expression, Lexical analysis, parsing

3. Paradigms of Programming language : **(10 L)**

Imperative paradigms (simple Procedural languages)Fortran, C. Block structured paradigm-,Pascal. The object based paradigm-Ada, C++, small talk, Logic programming paradigm-Prolog, The database language paradigm, Declarative paradigm, Event driven programming, Fourth generation languages

4. Building blocks of a language : (4 L)

Data Object , variable constants, Data types- A brief classifications, Derived, Abstract, User-Defined Type conversion

5. Procedures: (8 L)

Designing of procedure, Simple call return sub-program, Sub-programs, subroutines, Recursive subprogram, Referencing environment-local And global Different parameter passing call-By-value, call-by-reference

6. Functional programming: (8 L)

Characteristics of functional Programming, Elements of Functional Programming language, Functions, Functional forms, Functional declarations , Expression Evaluation Logic programming, Proofs, facts, queries, logical Variable, substitutions Instances.

7. Object oriented programming: (8 L)

Introduction Basic concepts-object, class, Characteristics of OOP, Abstraction, Encapsulation and information, Hiding, Inheritance ,Multiple inheritance, Polymorphism, OOP concepts in C++

8. Comparative study of programming languages: (2L)

Control structures Flow control, rules, Basic control structures

PGCS-103 : ADVANCED DATA STRUCTURES

(5Credits, 4L+2T)

Objectives:-

To study and understand the importance of efficient data structure in most software systems including operating systems, data bases, compilers and scientific simulations. Virtually all of the data structures are presented in the context of a single, unified, polymorphic class hierarchy.

This framework clearly shows the relationships between data structures how polymorphism and inheritance can be used effectively.

References:-

1)Data structures and algorithms in C++: Michael googrich & David Mount

2) Data structures and algorithms with object oriented design patterns: Bruno preiss

SYLLABUS:

1. **Foundation of data Structure**: **(4L)**
   Dynamic Arrays, singly linked lists, multidimensional Arrays

2. **Data Type & Abstraction:** **(2L)**
   Abstract data types, Design pattern

3. **Stacks**: **(6L)**
   concept, implementation using array, multiple stacks, applications

4. **Queues:** **(6L)**
   concepts, types of queue, implementation using array, applications

5. **Lists:** **(8L)**
   singly linked lists , doubly linked list, circular list, ordered lists, sorted lists

6. **Hashing:** **(4L)**
   basic idea, methods, .hash table, scatter tables, applications

7. **Trees:** **(8L)**
   concepts, n_ary trees, binary trees, Threaded Binary trees, traversal methods, expression trees, Implementation, applications

8. **Search table:** **(4L)**
   basics AVL trees, B_ trees, Applications

9. **Sorting :** **(6L)**
   merge sort, tree sort ,the set of ADT, quick sort, radix sort ,comparison

10. **Graphs:** **(8L)**
    the graph ADT, data structures for graph, graph traversal, directed graph, weighted graph, shortest path , Minimum spanning trees , applications

11. **Vectors:** vectors, lists, sequence **(4L)**

ELECTIVE I
PGCS-104: PARALLEL PROCESSING

(5Credits, 4L+2T)

Objective:

This subject aims at helping students to get the maximum benefit out of parallel & distributed computer systems. That is, this course aims at helping the student understand the interaction between hardware & software parts of the system, understand the power & limitations of parallel system, & understand the beneficial & challenging aspects of parallelism.

After completing this course, the student should be able to
1) Understand the fundamental aspects of parallel processing,
2) Be familiar with taxonomies of parallel system,
3) Be familiar with performance measures for parallel system,
4) understand the theoretical limitations of parallel computing,
5) Write efficient parallel application programs.

Text books:

1. Designing &building parallel programs (2004) by I. Foster. Addison Wesley, ISBN 0-201-57594-9
2. introduction to parallel computing , 2nd ed.(2003) by Ananth Grama, Anshul Gupta, George Karypis & Vipin Kumar. Benjamin/Cummings, ISSBN 0-8053-3170-0.

Reference Books:

3. Introduction to Parallel Processing by Prakash P. Ravi, Sasikumar M. & Shikhare Dinesh. Prentice Hall of India
4. Practical parallel programming by Wilson Gregory V. Prentice Hall of India
5. Parallel Programming : techniques & Application using networked Workstations & parallel computers, 2nd ed. (2004) by Barry Wilkinson & Michael Allen. Prentice hall, ISBN 0-13-140563-2.
6. Parallel computer architecture (1999) by D.Culler & J.P.Singh. Morgan Kaufmann publishers
7. Distributed & parallel computing (1997) by h. El-Rewini & t.g.Lewis
8. parallel programming with MPI(1996) by peter Pacheco Barnes & noble

SYLLABUS:

1 . Overview : **(8L)**

Motivation of parallelism, Scope of parallel Computing , Current trends in parallel processing

2. Concept Parallel Machine Model: **(6L)**

Parallel Programming Model , Parallel algorithm examples

3.Designing Parallel Algorithms: **(6L)**
Methodical Design ,Partitioning, Communication, Agglomeration, Mapping, Case Studies

4. Quantitative basis for design: Defining & Modeling performance, **(8L)**
Developing Models, Scalability analysis, Communication cost model ,Case Studies

5. Modular Designs: **(6L)**
Modularity & parallel computing Case studies

6 .Tools: **(12L)**
Compositional C++ ,C++ review ,C C++: Introduction C++: Communication
Case studies, Fortran M, Introduction , Communication ,Argument passing ,mapping , modularity, performance issues ,High performance Fortran, Data parallelism, data distribution & concurrency, Modularity, other HPF features & performance issues

7.Message passing interface: **(8L)**
MPI models & MPI basics, Communication, modularity & other features

8. Performance tools : **(2L)**
Performance analysis, Tools

9. Resources: Random nos.

10.Hypercube algorithm: Vector reduction & matrix transposition, Merge sort **(2L)**

11.Summary & revision: **(2L)**
Parallel & distributed computing, Programming & analytical tools

ELECTIVE I

PGCS-105 :THEORY OF AUTOMATA

(5Credits, 4L+2T)

1. **Objective:** The most fundamental component of computer theory is the theory of mathematical logic. Our subject is sometimes called computation theory rather than computer theory, because the items that are central to it are the types of tasks (algorithms or programs) that can be performed, not the mechanical nature of the physical computer itself.

2. **Reference Books:**

- Daniel LA Cohen, "Introduction to computer theory". Wiley Publication.
- John C. Martin, "Introduction to language and theory of computation", Mc Grawhill.
- HopCroft Ullman,"Introduction to automata Theory,Language and Computations",Narosa.
- Hapcraft Ulman "Introduction to Automata Theory".
- Harry R. Lewis,"Elements of Thoery Of Computation"

Other Reference Books that you may want to have a look are:

- Mishra and Chandrasekaran. "Theory of computer science", Automata languages and computation", New Delhi.
- E.V. Krishnamurthy "Theory of computer science" EWP publications.
- LIU.C.L., Elements of Discrete Mathmatics, Mc Grawhill.
- Aho,Ulman,Sethi "Principles of compiler construction"
- System Programming By John Donovan

SYLLABUS:

1. Introduction: (8L)

Review of Mathematical, Preliminaries, Relations, Functions, Set theory, and Predicate and Prepositional, Calculus, principle of Mathematical induction /strong, Mathematical induction

2. Finite state machine: (10L)

Definition, finite control, transition graphs, adjacency matrix, finite automata, determinist finite automata, language acceptance by FA, Moore and Mealy Machines, Finite state machine with output, FA MINIMIZATION AND RELATED THEOREM.

3. Regular Expression: (8L)

Recursive regular expression, regular set, NFA with E moves, NFA without E moves, Inter conversion between NFA and DFA, Regular expression and FA, pumping LEMMA

4.Grammer : **(10L)**

Context free grammar and it's properties, derivation tree, simplifying CFG, unambiguous CFG, CNF, GNF of CFG, chomsky hierarchy, chomsky normal form, derivation graphs type 0 and type 1 grammar. Concept of linear bounded automata, context sensitive grammars and their equivalence

5.Push down automata: **(8L)**

Push down automata 2 way PDA, relation of PDA with CFG, deterministic and nondeterministic PDA and related theorems

6.Turing Machine: **(10 L)**

Definition model, comparison of Turing machine, example of TM, universal TM, tm limitation, church's Turing hypothesis ,multi stack TM ,halting problem ,unrestricted grammars and their equivalence with TM, determinism and non-determinism of TM ,TM as acceptor /generator ,algorithms and related theorems multi-tape, multi-head, multi-stack TM

7.Complexity: **(4L)**

Introduction recursively numerable, sets, recursive set, partial recursive sets, Russell paradox undesirability and some non computable problems

8.Applications: **(2L)**

Application of RE and PA – lexical analyzer, text editor and searching using RE, application of PDA expression conversion, application on CFG, syntax analysis

ELECTIVE I

PGCS-106: DIGITAL IMAGE PROCESSING

(5Credits, 4L+2T)

Objectives

The objectives of this course are to:

- Cover the basic theory and algorithms that are widely used in digital image processing
- Expose students to current technologies and issues that are specific to image processing systems
- -Develop critical thinking about shortcomings of the state of the art in image processing

Reference books

1. Fundamental of Digital Image Processing by Anil k. Jain Prentice -Hall
2. Digital Image Processing by R.C. Gonzalez and Woods
3. Digital Image Processing using Matlab by R.C. Gonzalez and Woods,S .L Eddnins
4. Digital Image Processing PIKS Scientific Inside by William .K.Pratt ,WILEY India
5. Digital Image Processing and Computer Vision by Sonka ,Hlavac,Boyle, CENGAGE Learning

SYLLABUS:

1 What is Digital Image Processing: (4L)

Low level image processing, High level image processing, The origins of Digital Image Processing, Examples of fields that use Digital Image Processing, X-Ray Imaging, Imaging in the Ultraviolet Band, Visible and Infrared Band, Microwave Band, Radio Band, Fundamental steps in Digital Image Processing, Components/Elements of Digital Image Processing

2 Elements of Visual perception: (8L)

Structure of Human Eye, Image formation in the Eye, Brightness, Adaptation and Discrimination, Light and electromagnetic spectrum, Image sensing and acquisition, Image acquisition using a single sensor, Image acquisition using sensor strips ,Image acquisition using sensor array, A single image formation model, Image sampling and quantization, Basic concepts of sampling and quantization, Spatial and gray level resolution, Aliasing and moiré patterns, Zooming and shrinking of digital image, Some basic ,relationship between pixels, Neighbors of a pixel, Adjacency, Connectivity, Regions, Boundaries, Distance measures, Linear and non linear operations

3 Image Enhancement in the Spatial domain: **(10L)**

Introduction, Some basic grey level transformation Image negatives ,Log transformation, Power law(Gamma) transformations ,Piece wise linear transformation functions, Histogram processing, Histogram equalization ,Histogram matching(specification),Local enhancement, Image enhancement using arithmetic and logical operation,Image substation, Image averaging, Basics of spatial filtering, Smoothing spatial filters ,Order statistic (non linear)filters, Sharpening spatial filters, Use of second derivative for enhancement-the Laplacian, Use of first derivatives for enhancement – the gradient, Combining spatial enhancement methods

4 Image Enhancement in the Frequency domain: (10L)

Introduction to the Fourier Transform and the Frequency Domain, One Dimensional Fourier Transform,2-D Discrete Fourier Transform (DFT)and its Inverse, Filtering in the Frequency Domain, Correspondence between Filtering in the Spatial and frequency domain, Smoothing Frequency –Domain Filters

5 Image Restoration: (10L)

Introduction, Noise Models, Gaussian Noise, Rayleigh Noise, Erlang Noise ,Exponential noise, Uniform noise, Impulse Noise, Periodic noise, Restoration in the presence of noise only spatial filtering, Periodic noise reduction by frequency domain filtering, Band reject alters, Bandpass filters, Notch filters, Estimating the Degradation Function, Estimation of Degradation Function by Image Observation, Estimation of Degradation Function by Experimentation, Estimation of Image degradation by Modeling Geometric mean Filter, Inverse Filtering, Minimum Mean square error(Wiener)filtering ,Geometric Transformation

6 Morphological Image Processing: (8L)

Some basic concepts from set theory, Reflection and Translation, Logic operation involving Binary Images, Erosion and Dilation, Erosion, Dilation Duality, Opening and Closing, The Hit –or-Miss Transformation

7 Image Segmentation Fundamentals: (4L)

Detection of Discontinuities, Thresholding , Region –based Segmentation

8 Representation and Description: (6L)

Representation, Simple Boundary Descriptors, Simple Regional Descriptors, Use of principal components for description, Relational Descriptors

PGCS – 107 : LAB COURSE I

(4 Credits)

List of' programming assignments to be executed in C / C++:

1. Represent sparse matrix using array and perform matrix addition or simple and fast transpose.

2. Represent polynomial as a circular linked list and write a menu driven program to perform addition, multiplication and evaluation.

3. Write a menu driven program to perform following operations on doubly linked list : Create, Insert, Delete and Display.

4. Create two singly or doubly linked lists, sort them after creation using pointer manipulation. Merge these two lists into one list without creating, a new node. Merged list should be a sorted one.

5. Write a program to create a generalized linked list and perform following operations copy, equivalence and depth.

6. Implement Stack as an abstract data type using array or linked list. Use this ADT for expression conversion and evaluation.

7. Represent circular Queue using, array and write a pro-ram to perform following operations Insert, Delete, Finding front and rear element.

8. Creation of binary tree and perform recursive and non recursive traversals.

9. Creation of binary inorder threaded tree and perform all three traversals.

10. Represent a given graph using adjacency list and perform DFS and BFS.

11. Represent a (given graph using adjacency list or array and find the shortest path using Dijkstra algorithm.

12. Represent a given graph using adjacency list or array and (generate a minimum spanning tree using Kruskal's and Prime's algorithms.

13. Implement binary search tree as an abstract data type.

14. Create a binary search tree and find height of a tree and print the leaf nodes.

15. Create a binary search tree, find its mirror image, Print original and mirror image using levelwise printing.

16. Create a hash table and handle the collisions using linear probing with perform or without replacement.

PGCS – 108 : MINOR PROJECT-I

(4 Credits)

Objective : The objective of this minor project is to gear up student for preparation of final project training in Semester-IV.

Student will select individually Commercial or Technical project based on Technologies learnt in Semester I. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation. The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

PGGEN-12 : BASICS OF INFORMATION TECHNOLOGY

(2C, 30L)

Objectives: The main objective of this paper is to emphasize that system's thinking and abstract thinking are fundamental to efficient and effective software development..

Reference books

1. Rajaraman, V. (2004). Introduction to Information Technology. PHI.
2. Fundamental of Computers – By V. Rajaraman B.P.B. Publications
3. Computer Today- By Suresh Basandra
4. Unix Concepts and Application – By Sumitabha Das
5. MS- Office 2000(For Windows) – By Steve Sagman
6. Computer Networks – By Tennenbum Tata MacGrow Hill Publication
7. HTML Complete Reference

Chapter 1: Computer Fundamentals and Operating System (7 L)

Definition of a System, Elements of System, Characteristics of system, types of systems, Definition, Features of Operating System
Functions of Operating System, Types of O.S.-DOS, WINDOWS, LINUX

Chapter 2: Communication and Networking (7L)

Introduction, Network topology,
Types of Network
Intranet, Internet
OSI Model (Seven layers),TCP/IP
Transmission Media
Network Devices

Chapter 3 : Software Packages (8 L)

Microsoft Word – Mail merge
Microsoft Excel – Formulas, Graphs, Basis statistical formulae
Microsoft Power Point – Creating effective presentations
Microsoft Access –DBMS Features

Chapter 4 : Web Technology (8 L)

Basic Tags , Table layout , Form, Hyperlink
Frame layout, Scripting Languages**,** Features of JavaScript
DHTML

SEMESTER II

PGCS-201:SOFTWARE ARCHITECTURE

(5Credits, 4L+2T)

Pre-requisites:
Some familiarity with modern software engineering concepts and experience in designing and developing software systems.

Objectives:
This course provides a thorough overview of software architectures. This course will cover the scope of software paradigms with particular emphasis on software architecture. The students will learn concepts of software paradigms, software architecture and design patters.

Books:

1. Applied software architecture, C. Hofmeister, R. Nord, D. Soni, Addison Wesley, (2000)
2. Software paradigms, S.H. Kaisler, John Wiley & sons, (2005).
3. Design Patterns: Elements of Reusable Object - Oriented software, E. Gamma, R. Helm, R. Johnson, J. Vlissides, Addison – Wesley (1995).

SYLLABUS:

1.**Introduction :** **(8L)**
Basic Concepts, Introduction and need of Software Architecture , Software Architecture Design Plan, Software Architecture as an abstraction , Software Architecture terminology,Different Engineering concerns addressed by different views.

2. **Designing, Describing and using Software Architecture:** **(8L)**
System Overview,Product Features, System Interactions, The future of IS2000

3.**Global Analysis:** **(8L)**
Overview of Global Analysis Activities, Analyze factors, Develop strategies,
Analyze Organisational , Technological, Product Factors, Global Analysis Summary

4.**Conceptual Architecture view:** **(8L)**
Design Activities for the conceptual, Architecture view, Design of Conceptual view for IS2000,Summary of Conceptual Architecture view ,Traceability. Uses for the Conceptual Architecture view.

5.**Module Architecture View:** **(8L)**
Design Activities for the Module Architecture view, Design of Module view for IS2000. Uses for the Module Architecture view.

6.**Execution Architecture View:** **(6L)**
Design Activities for the Execution Architecture View ,Design of Execution View for IS2000.Design summary for IS2000 Execution View, Summary of Execution Architecture View

And Traceability. Uses for the Execution Architecture View.

7. **Code Architecture View : (6L)**

Design Activities for the Code Architecture View, Design of code Architecture View for the IS2000.Summary of code Architecture ,View and Traceability ,Uses for the code Architecture View

8. **Software Architecture Best Practice: (4L)**

Global Analysis, Conceptual Architecture View ,Module Architecture View, Execution Architecture View, Code Architecture View, Software Architecture Uses

9. **Role of software Architecture: (4L)**

Creating a Vision, The Architect as a Key Technical Consultant, The Architect makes decisions, The Architect Coaches, The Architect Coordinates, The Architect Implements

PGCS- 202: NETWORK SECURITY

(5Credits, 4L+2T)

1. **Objective:** Information is an important strategic and operational corporate asset. Computer and computer network, which are increasingly being used for storing and retrieving information, consequently need to have adequate security measures that can safeguard sensitive information. This course introduces the student to the rapidly emerging area of computer and network security.

2. **Reference Books:**

Charlie Kaufman, Radia Perlman, Mike Speciner, "NETWORK SECURITY- Private Communication in a PUBLIC WORLD", 2nd Edition, Prentice Hall.

Other Reference Books that you may want to have a look are:

1. Edward Amoroso, "**Fundamental of Computer Security Technology**", Prentice Hall.
2. William Stallings, "**Cryptography and Network Security-Principle and Practice** ", Prentice Hall.
3. Marsall D.Adams, Suil Jajodia and Harold J. Podell, Eds, "**Information Security –an Integration Collection of Essays**", IEEE Computer Society Press.
4. William R. Cheswick and Steven M. Bellowin, "**Firewalls and Internet Security, Repelling the Wily Hacker**", Addision-Wesley.
5. Gunter Schafer.**Security in Fixed and Wireless networks**.John Willy & Sons.
6. J.Schiller."**Mobile Communications**",Second edition,Addison-Wesley,2003
7. Charles P.Pfleeger,"**Security in Computing**",Prentice Hall.
8. Warwick Ford,"**Computer Communication** Security",Prentice Hall.
9. William Stalling,"**Network Security Essentials**",Prentice Hall.

SYLLABUS:

1. Introduction : (10L)

OSI Reference Model, IP, UDP, and TCP, Replicated Directory Services ,Packet Switching, Network Components, Destination: Ultimate and Next Hop. Address Structure, Active Vs Passive Attacks, Viruses, Worms, Trojan, Horses, Multilevel Model of Security

2.Introduction to Cryptography: (8L)

What is Cryptography, Breaking an Encryption scheme, Types of Cryptography, Functions, Secret Key Cryptography ,Public Key Cryptography, Hash Algorithms

3. Secrete Key Cryptography: (12L)

Generic block Encryption ,Data Encryption standard (DES),International Data Encryption Algorithm (IDEA),Advanced Encryption Standard (AES), Modes of Operation, Encrypting Large Message, Generating MAC's, Multiple Encrypting DES, Hashes and Message Digests an Introduction

4. Public Key Algorithm: (8 L)

Modular Arithmetic, RSA Algorithm, RSA Algorithm, Diffe-Signature standard (DSS),Number Theory, AES and Elliptic Curves

5. Authentication System: (12 L)

Password-Based Authentication, Address-based Authentication ,Cryptographic Authentication Protocols, Eavesdropping and Server Database Reading, Trusted Intermediaries, Session Key Establishment, Authentication of people, Security Handshake Pitfalls

6. Electronic Mail Security: (7 L)

Distribution List, Security Services for Electronic mail, Privacy Authentication of sources, Message Integrity, Proof of Submission and Delivery, Message Flow Confidentiality, PEM and S/MIME,PGP (Preety and Good Privacy)

7. Firewalls: (3 L)

Packet Filters, Application Level Gateway, Encryption Tunnels, Security System**,** NetWare V3,V5, DCE Security

PGCS-203: JAVA APPLICATION PROGRAMMING

(**5Credits, 4L+2T)**

Pre-requisites

A] participants should be familiar with basic programming concepts, basic objects oriented programming concept with java

Objective

Course designated to provide programming skills needed for developing disrupted and web application.

Books:

I. core java vol -II CAY R. Horstman and Gray cornel, Pearson Education
II. Java Server programming, wrox press
III. Inside Servlet, Dustin R. Callway, Pearson Education
IV. Entrprise JavaBeans, Richard Monson Hafel, Oreilly Publication

SYLLABUS:

1.Networking : (9L)

networking basics, Connecting to the server, sockets for client and server one example writing client and server, UDP Datagram socket

(11L)

2.Java data Base :

JDBC Introduction, Types and drivers, Querying databases, JDBC Metadata, Creating prepared statements, Scrollable result sets, Transaction management, connecting pooling, save points and performing batch updates, distributed Transaction support

3 .Java remote Method Invocation(RMI): (10L)

introduction to distributed computing using RMI, RMI Architecture, Writing simple RMI application ,Invoking remote object ,Object serialization for remote parameters

4.Servlet and JSP: (10L)

Introduction and life cycle, Servlet API, Servlet and Thread safety, HTTP Redirect, Managing User state: code, Session Tracking, servlet and JDBS, servletside Includes, JSP Introduction $ JSP Directive, JSP Scripting elements JSP Standard Action, JSP Implicit Object

5.Enterprise Java Beans: (10L)

introduction to EJB, Session beans, Entity beans(CMP/BMP)JNDI, One example development of enterprise,JavaBeans, Connecting EJB to web module, Enterprise application design consideration, Application deployment

6. MVC Architecture and Struts Framework: (10L)

introduction to MVC architecture, Development of controller Servlet, Development of controller Servlet, Development of Model bean, Development of view components, Component integration and deployment, Introduction to struts, Struts application framework, struts controller model and view components, JSP Custom tag library, Struts and EJB

ELECTIVE II
PGCS-204: SOFTWARE TESTING

(5Credits, 4L+2T)

Objectives :

To discuss the distinctions between validation testing and defect testing.
To describe the principles of system and component testing.
To describe strategies for generating system test cases.
To understand the essential characteristics of tool used for test automation.

Pre - requisites:

Some familiarity with modern software engineering concepts and experience in designing and developing software systems.

Reference Books:-

- Software Testing: Ron Patton
- Effective Methods for Software Testing: William E Perry, Wiley
- Software Testing in the Real World: Edward Kit
- Introducing Software Testing: Louise Tamres
- Software testing Techniques: Boris Beizer
- Effective Methods for Software Testing: WilliamPerry, Wiley
- Software Testing: Renu Rajani

SYLLABUS:

Chapter 1: Big Picture: Introduction: (6L)

Infamous case studies. What is a bug? Software failure terminologies, Why do bugs occur? Cost of bugs in software, Categories , Consequences ,Testing Fundamentals. Test Axioms. Six Essentials of testing, Tester: ? Competencies. Assessing Quality of the organization.

Chapter 2: The S/W Development Process: (6L)

Realities of S/W testing, Life Cycle Models, Building a s/w testing process. Realities of S/W testing, Types of testing. Testing terms and definitions.Developing a test plan, Requirements verification checklist. Case study, Workbench, Do Procedures, Check Procedures.

Chapter 4: **(6L)**

Testing Methods, Verification Testing, Validation Testing, What are test cases? Creating test cases. Test case planning overview, Reducing the number of test cases.

Chapter 5: (6L)

Eleven Step Testing Process, Testing Process Overview: Clean Sheet Approach, Eleven Steps, Issues in requirements testing. Black-box, White-box, Static and Dynamic Testing.

Chapter7: **(6L)**

Examining code, Issues in Code Testing. Static methods ,Case Study, Testing Software with X-Ray glasses, Dynamic testing, Coverage Types. Validation, Validation Testing, Controlling validation Costs, Alpha & Beta Testing, Software Testing tools, Types continued, Need for Tools, Types.

Chapter 8: **(6L)**

Test Documentation & Reporting, Bug Reporting, Bug Tracking Systems, Measurement Types. Metrics, Recommendations. Types of testing: More categories, Flow Testing, Path testing, Syntax Testing, Transition Testing, Compatibility Testing, Usability Testing, Configuration Testing, Web Site Testing, Foreign Language Testing.

Chapter 9: **(6L)**

Software Quality Assurance ,CMM, Career: Software Tester,Hands-On Experience. Training. Professional Organizations, Managing Test Technology.

Chapter10: **(6L)**

Current Practices & Trends, What is new? Developer to Tester Ratio? Benchmarks, Test Security. Testing off-shelf s/w, Identification and precautions.

ELECTIVE II

PGCS-205:EMBEDDED COMPUTING

(5Credits, 4L+2T)

1)Objective:

Introduce the students to the concepts and practices involved in the embedded computing domain.

2)Reference Books:

1: Dreamtech Software Team, 'Programming For Embedded Systems', Wiley-dreamtech India Pvt.
2: Lewis D.W., 'Fundamentals of Embedded software: Where c and assembly Meet', Pearson Education Asia

SYLLABUS:

1. **An overview of embedded computing : (8L)**
   Introduction to embedded systems, Categories of embedded systems, Requirements of embedded systems, Trends in embedded software
Development

2. Applications of Embedded Systems: (12L)
Consumer electronics, Control systems and industrial Automation, Biomedical systems, Field instrumentation, Handheld computers, Data communication, Network information appliances, Telecommunications, Wireless communications

3.Hardware architectures For embedded systems: (10L)
Hardware architecture, Processor, memory, latches and buffer, crystal, reset-circuit, chip-select logic circuit, ADC and DAC, application-specific control circuitry, display units, keypads Development tools: EPROM programmer, EPROM eraser

4.Developing an embedded system: (10L)
Determine the requirement, Design the system architecture, Choose the operating system, Choose the processor, Choose the development platform, Code optimization, Verification of the software on the host system,

5.Embedded software development environment: (10L)
Operating systems: Embedded operating systems, Real-time operating systems, Programming Languages: Assembly Languages, high-level languages.

6. I/O interfacing in embedded systems (10L)
The Intel I/O instructions, Synchronization, transfer rate, and Latency, Polled waiting loops, Interrupt–driven I/O, Direct memory access

7.Communication Interface standards in Embedded systems: (10L)
Serial interface/ UART,IEEE 1394,Universal Serial Bus, IrDA, Ethernet, Bluetooth, PCI bus

ELECTIVE II

PGCS-206: SOFTWARE PROJECT MANAGEMENT

(5Credits, 4L+2T)

Objectives:-

- To Articulate similarities and differences between IT projects and other types of projects.
- Apply general project management competencies to IT projects.

- Apply the techniques and develop the documents related to IT project management.
- Understand how to apply different life-cycle models t design IT projects.
- Understand the nature of projects that plan plan-driven and agile development
- Methodologies.
- Identify IT project risks and develop risk mitigation strategies.
- Identify cases of IT project escalation and de-escalate troubled IT projects.
- Understand how the organizational environment can facilitate IT project success.
- Understand how to facilitate learning in and learning from IT projects.

References:

1. Robert T.Futrell, Donald F. Shafer, Linda I. Safer, "Quality software project management", Pearson Education, Asia, 2002.
2. Pankaj Jalote,"software project, manegment in practice", Addison Wesley, 2002
3Hughes," Software project manegment,3/E",Tata McGraw-Hill,2004

SYLLABUS:

1.Introduction to competencies: (12L)

product development technique - management skills - product development life cycle - software development process and models - The SEI CMM - international organization for standardization

2. Domain process (12L)

Managing domain process project selection models-project portfolio management – financial process selecting a project team – goal and scope of software project - project planning - creating the work breakdown structure – approaches to building a WBS – project milestones – work packages-building a WBS for software

3. Software development: (12L)

Tasks and activities – software size and reuse estimating- the SEI CMM – problems and task - cost estimation - effort measures - COCOMO: A regret ion model – COCOMO II – SLIM: a mathematical model - organizational planning - project roles and skills needed.

4. Scheduling activities: **(12L)**

Project management recourse activities - organizational form and structure –software development dependences - brainstorming – scheduling fundamentals - PERT and CPM – leveling recourses assignment-ma the schedule to a real calendar – critical chain scheduling

5. Quality assurance **(12L)**

Quality requirements – the SEI CMM – guideline – challenges - quality function deployment – building the software quality assurance – plan –software configuration management: principals – requirements - planning and organizing – tools – benefits - legal issues in software - case study

PGCS-207: LAB COURSE II

(4 Credits)

1.Java program to create objects of a class .

2.java program to show constructor overloading.

3.java program to compare two stack variables

4.java program to show private member of super class

5.java program to show inheritance

6.java program to show polymorphism

7.java program to calculate charges of various trunk calls using polymorphism

8.java program to import package and access method in it

9.Java to show the usefulness of Interfaces as a place to keep constant value of the program.

10.java program to show method overriding

11.java program to show implementation of interface.
12.java program to show Array index out of bound Exception

13.Java program to show divide by zero exception.

14.java program to throw number format exception

15.java program to avoid vehicle collision of vehicle coming in opposite direction

16.java program to show thread priority

17.java program to launch multiple threads and count them.

18.java program to generate 2 threads, one for printing even numbers and the other for printing odd numbers.

19.javaprogram to show thread synchronization.

20.java program to read text and display length in textbox

21.Write a program in Java for String handling which performs the following:
i) Checks the capacity of String Buffer objects.

ii) Reverses the contents of a string given on console and converts the resultant string in upper case.
iii) Reads a string from console and appends it to the resultant string of ii.

22.java program to find first and last occurrence of char substring in a string

23.Java program to read a statement from console, convert it into upper case and again print on console.

24.java program which takes the name of a file from user, read the contents of the file and display it on the console

25.Java program to copy a file into another file.

26.java applet to read text and display length in textbox use colors
java applet which displays a rectangle/string with specified colour & coordinate passed as parameter from the HTML file.

27.java applet which will display the calendar of a given date.

28.java program to display student's detail using Card Layout
java applet to add scrollbars to text area and find a word in a text.

29.java applet to accept name age and display message and url

30.javaprogram to test Socket functionality for appropriate hostname and port number.
31.Java program to connect to a database and perform operations of Selection, Insertion and Deletion on the database.

PGCS – 208: MINOR PROJECT

(4 Credits)

Objective :

The objective of this project is to gear up student for preparation of final project. in Semester-IV. Student will select individually Commercial or Technical project based on Technologies learnt in Semester I,II.

Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies. Student may use any Design Methodologies such as SSAD, OOAD and UML etc.

This is a documentation project only. The project work will be presented by student using Power Point Presentation . The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

PGGEN23: Cyber Security

(2C, 30L)

Objective : To Learn Cyber Security – threats, detection and prevention and Cyber Laws and provisions.

Reference:

1) Cyber Law in India by Farooq Ahmad – Pioneer Books
2) Information Tecnology Law and Practice By Vakul Sharma – Universal Law Publishing Co. Pvt. Ltd.
3) The Indian Cyber Law by Suresh T Vishwanathm – Bharat Law house New Delhi.
4) Hand book of Cyber & E-commerce Laws by P.M.Bakshi & R.K.Suri – Bharat law house New Delhi.
5) Guide to Cyber Laws by Rodney D. Ryder – Wadhwa and Company Nagpur.
6) The Informantion Technology Act,2000 – Bare Act – Professional Book Publishers – New Delhi

SYLLABUS :

Chapter 1 : Objectives and Encryption (5 L)

Object, Scope of the Act, Symmetric Cryptography, Asymmetric Cryptography, RSA Algorithm, Public Key Encryption

Chapter 2 : Digital Signature (3 L)

Technology behind Digital Signature, Creating a Digital Signature, Verifying a Digital Signature

Chapter 3 : E-Governance and IT Act 2000 (5 L)

Legal recognition of electronic records, Legal recognition of digital signature, Use of electronic records and digital signature in Government

Chapter 4 : Certifying Authorities (7 L)

Need of Certifying Authority and Power, Appointment, Function of Controller, Who can be a Certifying Authority?, Digital Signature Certifications, Generation, Suspension and Revocation of Digital Signature Certificate

Chapter 5 : The Cyber Crimes (S-65 to S-74) (7 L)

Tampering with Computer Source Documents (S-65), Hacking with Computer System (S-66), Publishing of Information Which is Obscene in Electronic Form(S-67), Offences : Breach of Confidentiality & Privacy (S-72), Offences : Related to Digital Signature Certificate(S-73 & S-74)

.

SEMESTER III

PGCS-301- ARTIFICIAL INTELLIGENCE

(5Credits, 4L+2T)

Objectives:

To understand and apply principles, methodologies and techniques in design and implementation of intelligent system.

References:

1. Struart Russeland Peter norvig, .Artificial intelligence: A modern Approach.
2. George F. Luger, . Artificial intelligence: Structures and Strategies for Complex Problem Solving., Person Education
3. Nils J. Nillson, .Artificial Intelligence : A New Synthesis., Harcourt Asia page 36 of 38
4. Elaine Rich and Kevin Knight,. Artificial Intelligence.,TMH
5. Patrick Winston,. Artificial Intelligence., Pearson Education
6. Ivan Brakto,.Prolog Programming for Artificial Intelligence.,Pearson Education
7. Efraim Turban Jay E.Aronson, .Decision Support System and Intelligent Systems.
8. Ed. M. Sasikumar and Others, . Artificial Intelligence :Theory and Practice.

Proceedings of the International Conference KBCS-2002, Vikas Publishing House

SYLLABUS:

1. **Artificial Intelligence:**
An overview, Intelligent Systems: Evolution of the concept. **(2L)**

2**Intelligent Agents** :
how agent should act, Structure of intelligent agents, Environments **(3L)**

3 **Problem solving**
solving problems by searching, informed search methods, Game playing **(5L)**

4**Knowledge and Reasoning** **(7L)**
A knowledge based agent, Representation, Reasoning Logic, Proportional logic, First order logic: Syntax and Semantics, Extensions and Notational variation ,Using first order logic

5**Building a Knowledge Base**
Properties of good and bad Knowledge base, Knowledge Engineering, general ontology (4**L)**

6 **Interfacing first Order Logic:**
Interface rules involving quantifiers, An example proof, Forward and backward chaining, Completeness **(5L)**

7**Acting logically**
Planning, Practical planning: Practica l Planners, Hierarchical decomposition, Conditional planning **(4L)**

8 **Uncertain Knowledge and Reasoning:**
Uncertainty, Representing knowledge of uncertain domain, the semantics of belief networks, Inference in belief networks **(4L)**

9**Learning:**
Learning from observations: General model of learning agents, Inductive learning, learning decision tree, Learning in neural and belief networks: introduction of neural networks, Perceptrons, Multilayer feed forward networks, Application of ANN, Reinforcement learning: Passive learning in a known environment, Generalization in reinforcement learning, Genetic algorithms **(7L)**

10**Agents that Communicate:**
Communication as action, Types of communicating agents, A formal grammar for a subset of English **(2L)**

11**Expert system:**
Introduction of expert system, Representing and using domain knowledge, Expert system shells, Explanation, Knowledge acquisition **(3L)**

12**Application:**
Natural language processing, perception, Robotics **(2L)**

PGCS-302- MOBILE TECHNOLOGIES

(5Credits, 4L+2T)

Objectives:

1. To familiarize the students with the buzz words and technology of mobile communication.
2. Understand the GSM architecture.
3. Understand the issues relating to Wireless applications.

Reference Books –

1) Mobile Communication By Jochen Schiller
2) Pervasive Computing

SYLLABUS:

1. Introduction to Mobile Computing: **(2L)**
Introduction and need for Mobile computing, Mobility and portability, Mobile and Wireless devices, Applications, Brief History of wireless communication

2 Wireless Transmission: **(3L)**
General Concepts of multiplexing and modulation, Spread Spectrum, Cellular Systems, Cellular Phone Array, Mobile Phone Technologies (1G, 2G, 2.5G, 3G)

3.Medium Access Control Layer: **(4L)**
Why specialized MAC?, hidden and exposed terminals, near and far terminals, General Concepts and comparison of SDMA, FDMA, TDMA ,CDMA

4. Global System for Mobile Communication: **(9L)**
mobile Services (Bearer, Tele-and-supplementary services), System Architecture, Radio subsystem, Network and switching subsystem, Operation subsystem, Protocols, Localization and calling, Handover, Value Added Services, SMS, Architecture, Mobile Originated and Mobile Terminated, procedures, Cell Broadcast Service, Architecture, Message Transfer Procedure, MMS, Architecture, Protocol framework, Message Transfer, Procedure, Location Services, Logical Reference Model, Control procedures, Network Architecture, determination of Location Information, Location based services, GPRS

5.**Mobile IP :** **(10L)**
Goals, assumptions and requirements, Entities and terminologies, Agent Discovery, Registration, Tunneling and encapsulation, Reverse Tunneling, IPv6, IP micro-mobility support – Cellular IP, Hawaii, Hierarchical, mobile IPv6, Mobile Routing Destination sequence distance Vector, Dynamic Source Routing, Alternative Matrix, Adhoc Routing Protocols -Flat, Hierarchical, Geographic-position-assisted

6 **Mobile TCP:** **(5L)**
Traditional TCP, Congestion Control, Slow start, Fast retransmit / Fast recovery, Implications on mobility, Classical TCP improvements, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit / Fast recovery, Transmission / Timeout freezing, Selective Retransmission, Transaction oriented TCP, TCP over 2.5/3G wireless networks

7.**Wireless Application Protocol** : **(4L)**
Architecture , Wireless datagram protocol, Wireless transport layer security, Wireless transaction protocol, Wireless session protocol, Wireless application environment , WML, WML Scripts, Push Architecture, Push – Pull Services

8. **Java for Wireless Devices:** **(2L)**
Setting up the development environment, Basic Data types, Libraries (CLDC, MIDP)

9. **UI Controls:** **(3L)**
Displayable and Display Image, Events and Event Handling, List and choice, Text box, Alerts

10. **Persistent Storage:** **(2L)**
Record Stores, Records, Record Enumeration

11. **Network MIDlets:** **(2L)**
The Connection Framework, Connection Interface, Making a connection using HTTP, Using datagram connection

12. **Wireless Messaging:** **(2L)**
Architecture for Messaging application, Messaging API, Types of applications, Pro’s and con’s of messaging

PGCS-303: .Net Technologies

(5Credits, 4L+2T)

Objectives:-

Student can do independently small projects in Windows Applications (Vb.net).He will also have primary knowledge of Web Applications (ASP.Net).He will have introductory of C# so within 1 or 2 weeks he can do the projects in C#.

Reference:-

1.MCAD / MCSD:Developing Windows-Based Applications With Microsoft Visual Basic.Net and Visual C# .Net By Mattjew Stoecker

2.MCAD / MCSD:Developing Web-Based Applications With Microsoft Visual Basic.Net and Visual C# .Net By Jeff Webb

3.Programming Microsoft VB.NET Microsoft Press By Belena

4.Programming Microsoft VB.NET ASP.NET By Dino Esposito

SYLLABUS

1Introduction to .Net Technology: (2L)

Framework common Language Runtime, Common Language Specification, Intermediate Language Code, Just-In-Time Compiler, Assemblies, Manifest, Metadata ,Global assembly Cache, VB.NET, Creating Applications, Building, Running Projects, Data , Operator, Loops, Procedures, Functions.

2. Windows Forms: (5L)

Text Box, Buttons, Labels, Checks Boxes, radio Buttons, List Boxes, Combo Boxes, Picture Boxes, Scrollbars, Timer, Menus, Built-in Dialogs, Image List, Toolbars, Status Bar and Progress bars, Event and Delegates, Tracing, Debugging

3Object Oriented Programming in VB.NET: (5L)

Class and Objects Properties, methods and events, Contractor and Destructor, Method overloading, Inheritance, Access modifiers, Public, Private, Protected, Friend, Overriding and shadowing, Interfaces, Polymorphism, Error Handling, Private and Shared Classes

4File Handling: (3L)

Stream Writer, Stream Reader, MSDN, Binary Reader, Binary Writer Classes MSDN, File and Directory Classes

5.ADO.Net Connections : (8L)

Data adapters, and Datasets ,Data binding with controls, Navigating data source ,Data from wizard, Data reader, Connection objects, command Objects, Data Adapters, Dataset Class, Data Readers, Non-Queries, Query-Parameters, Scalar-Queries, Calling stored-Procedures, Data-Relations, Editing rows ,Filtering and sorting, XML (Reading, Writing)

6Web application: **(5L)**
Introduction to Web form, page directive, all validation controls, Page redirection

7 State management: **(5L)**
Application state, Session State, View state, Web controls, Tracing web applications

8Database: **(2L)**
Data grid control in web applications

9 Web services: **(5L)**
Concept of web services, MSDN, Create a small web services

10Deployment: **(5L)**
Deploying applications using wizard MSDN

11Security introduction: **(5L)**
Securing a Microsoft Applications, MSDN

12C#: **(5L)**
Introduction to C#, Windows Forms with C#, Difference between C# and VB.Net MSDN
Textbox, Label, Command button

13Object Oriented Programming in C#.Net: **(5L)**
Class and Objects, Properties, Methods and events, Contractors and Destructor, Method overloading, Inheritance, Interface

ELECTIVE III
PGCS-304 : Soft Computing and Data Mining

(5Credits, 4L+2T)

Objectives : 1. Makes students familiar with the concept of data mining and its applications in the real world.
2. To be able to mine data and find patterns from that data.

Reference Books:-

1. Jiawei Han, micheline Kamber,"Data mining concepts and Techniques", Morgan Kaufmann publishers,2002.
2. Alex Berson,Stephen J.smith,"data warehousing, data mining and OLAP" Tata McGraw Hill 2004
3. M.H. Dunham "data mining" Pearson education.

SYLLABUS:

1. Data warehousing: (10L)
Need for data warehousing , architecture of DW, benefits of DW,OLAP and data cubes, data preprocessing – need, data cleaning, data integration and transformation, data reduction, roll up ,drill down.

2. Data marts: (4L)
Definition, reasons for creating data marts, designing data marts.

3. Introduction to Data Mining: (8L)
Definition, basic data mining tasks, knowledge discovery in databases, issues in the data mining, applications of data mining.

4. Data mining techniques: (8L)
Association rules, frequent item sets and association rule mining: apriori algorithm, FP growth algorithm.

5. Classification: (13L)
Definition, need of classification, decision tree learning, Bayesian classification, Naive Bayes classifier, linear classifiers, linear regression, and non linear regression.

6. Clustering: (12L)
Definition, need of clustering, types of clusters, similarities and distance measures, partitional algorithms like nearest neighbor algorithm- means.

7. Applications of data mining: (5L)
Social impacts of data mining, mining text database, mining spatial databases, mining web data.

ELECTIVE III

PGCS-305 : Advanced Database Management systems

(5Credits, 4L+2T)

Objective : **1)**Be able to translate complex conceptual data models into logical and physical database designs.
2) Be able to design high-quality relational databases and database applications.
3)To understand location ,replication and fragmentation independence.
4)To Expose the students to advance database concepts and techniques.

Reference Books:-

1. Fundamentals of Database Management Systems By Navathe & Elmasri (3rd Edition, Pearson Education)
2. Principles of Distributed Database Systems by Patrick Valduriez (3rd Edition' Pearson Education)
3. Introduction to Database Systems by C G Date (7th Edition Pearson Education)

SYLLABUS:

Object Oriented Databases:- **(5L)**

Overview of Object Oriented Concept ,Object Identity ,Object Structure , Object Definition Language, Types of Constructors , object database conceptual design, OODBMS advantages, Object query language examples of OODBMS

Distributed Databases :- **(25 L)**

Introduction to Distributed data processing, Homogeneous and heterogeneous systems ,Distributed DBMS Architecture, fragmentation , Distributed database design , Overview of Query processing, Query decomposition and data localization

Distributed DBMS reliability:- **(15 L)**

Reliability concepts & measures, Failures & fault tolerance in distributed systems, Local reliability protocols , Distributed reliability protocols, Network partitioning

Parallel Databases:- **(10 L)**

Parallel database concepts , Parallel database system architecture, Query parallelism, parallel Data processing, parallel query optimization

Emerging Database Technologies & Applications:- **(5 L)**

Multimedia Databases , Mobile Databases, Geographical Information Systems , Spatio – temporal patterns, Intervals and scalar operators in temporal databases, web databases,
Deductive Databases

ELECTIVE III

PGCS-306 : ADVANCED OPERATING SYSTEM

(5Credits, 4L+2T)

References:-

1. Operating Systems – Internals and Design principles by William Stallings (Fifth Edition, Prentice Hall India – 2005)
2. The MINIX Book: Operating System – Design and Implementation by Andrew S Tanenbaum and Albert S Woodhull(Third Edition, Prentice Hall India – 2006)
3. UNIX Systems for Modern Architectures by Curt Schimmel, Addison Wesley
4. Applied Operating Systems Concepts, Seventh Edition by Abraham Silberschatz, Peter Baer Galvin, and Greg Gagne, John Wiley and Sons, 2004

SYLLABUS:

1.Overview of Operating systems: (3L)

Objectives and Functions, Essentials of modern operating systems, Overview of UNIX, LINUX and Windows

2.**Process Management and Scheduling: (7L)**

Process and its states, Process control, Scheduling algorithms, Multi-processor and real time scheduling, UNIX Process management & scheduling, Windows Scheduling

3 **Threads and Symmetric Multi-processing (9L)**

Processes and Threads, SMP, Implementation of above in Popular Operating Systems

4 **Concurrency Control – Mutual exclusion and synchronization, Deadlocks and Starvation (9L)**

Mutual Exclusion & concurrency, Semaphores, Monitors, message Passing, Readers/Writers problem

5.**Memory Management and Virtual Memory (9L)**

Memory Management ,Partitioning, Paging and Segmentation ,Virtual memory and associated algorithms ,Implementation of above in Windows and UNIX

6 **I/O management and File management (9L)**

I/O devices, I/O functions, I/O buffering, Disk Scheduling, RAID levels & Disk cache ,Virtual File Systems, UNIX and Windows support for I/O, File Organisation, directories, sharing, Storage management and Record Blocking, UNIX and Windows File management

7 **Networking and Security: (6L)**

Protocol architecture, TCP/IP, Sockets, Network File System, Security Threats, protection, Intruders, malicious software, trusted systems

8 **Distributed processing, client/servers and Clusters (8L)**

Client/server computing, message passing, RPC Clusters

PGCS - 308: Lab Course III

(4 Credits)

Using Controls -

1. Create basic calculator utility in vb.net.
2. Write vb.net program using trimmer & progress bar controls & set their properties.
3. Write a program in vb.net using list box control & create add, remove & clear utility.
4. Write a program in vb.net that demonstrates use of check box control.
5. Write a program in vb.net that demonstrates use of radio buttons.

Using OOP concepts –

6. Write program in vb.net for implementing simple inheritance.
   -Create employee as base class & inherit it in manager derived class.
7. Write program in vb.net for implementing multiple inheritance.
8. Write program in vb.net for implementing abstract class.
9. Write program in vb.net for implementing interface, polymorphism
10. Write program in vb.net for implementing Abstract class.
    - Create vehicle, motor & car classes.
    - Create vehicle as abstract class.
    - Inherit abstract class into motor class & car class.

Using ado .net-

11. Write program in vb.net for create ado.net connectivity using connected architecture. (using Connection, Data reader , command object) & display data in List box.
12. Write program in vb.net for create ado.net connectivity using disconnected architecture.(using Connection ,Data Adapter ,Data dataset object) & display data in textboxes & navigate data. (first ,prev, next, last)
13. Write vb.net program using ado.net for multiple table connection through wizard. Display data in textbox, list box & data grid control.
14. Create crystal report in vb.net.

Using Exception handling and File handling-

15. Write vb.net program for exception handling. (Try-Catch)
16. Write program for File handling in vb.net. writing & reading text file using stream reader & stream writer class.
17. Write program for File handling in vb.net
18. Writing & reading binary file using binary reader & binary writer class.

PGCS – 308: MINOR PROJECT- III

(4 Credits)

Objectives :

The objective of this minor project is to gear up student for preparation of final project training in Semester-IV. Student will select individually Commercial or Technical project based on the technologies learnt in Semester I,II & III. Each student will have to prepare proper documentation consisting of SRS, Modeling Techniques, Development Strategies and Implementation and Testing Strategies.

Student may use any Design Methodologies such as SSAD, OOAD and UML etc. This is a documentation project only. The project work will be presented by student using Power Point Presentation . The Institute may appoint external expert from industry or academics if it feels so. The students will be assessed internally by such panel for this project.

SEMESTER IV

PGCS – 401: INTERNSHIP

(16 Credits)

Objectives:-

Fourth semester is internship for 200 marks. Project work may be done individually or in groups in case of bigger projects. However if project is done in groups, each student must be given a responsibility for a distinct module and care should be taken to see the progress of individual modules is independent of others.

Students should take guidance from an internal guide and prepare a Project Report on "Project Work" in 2 copies to be submitted in the department by first week of march. The Project Report should contain an Introduction to Project, which should clearly explain the project scope in detail. Also, Data Dictionary, DFDs, ERDs, File designs and a list of output reports should be included.

Project work can be carried out in the Institute or outside with prior permission of the Institute. Project viva-voce by the University panel will be conducted in the month of March-April.

General Instructions Regarding Preparation of Project Report for M.Sc.(Comp. Sci.) SEM-IV

TYPING

1. The typing shall be standard 12 pts in double spaced using black ink only
2. Margins must be Left 2 inches Right 1.5 inches Top 2 inches Bottom 1.5 inches
3. Paper A4 size Bond Paper

COPIES

Two hard-bound copies
(Black Rexine with Golden Embossing as per format displayed herewith)
One original and one clean Xerox Copy.

Project Evaluation Phases :-

Recommended Phase	Description	Timeline
1	SRS Document	3nd Week
2	Design document	7th Week
3	Executable/User Interface	12th Week
4	Test plan and Documentation	16th Week
5	Project Viva/Presentation	20th Week