SBRR MAHAJANA FIRST GRADE COLLEGE (Autonomous)

POST GRADUATE WING
(Accredited by NAAC with ‘A’ grade)
Pooja Bhagavat Memorial Mahajana Education Centre.
Affiliated to University of Mysore.

M.Sc. in Computer Science

Motto: Enter to Learn, Depart to Serve.

Vision:

To Build a strong research and teaching environment that responds swiftly to the challenges of the 21st century.

The **Mission** of the Computer Science Department is to:

1. Provide the highest quality education in Computer Science;
2. Perform research that advances the state-of-the-art in Computer Science;
3. Produce post graduates who are knowledgeable, articulate, principled, innovative, confident, and able to think critically;
4. Be engaged in local, State, and National issues to the benefit of both public and the private sector; and
5. Maintain a diverse college community.

SBRR MAHAJANA FIRST GRADE COLLEGE (Autonomous)

POST GRADUATE WING
(Accredited by NAAC with 'A' grade)

Pooja Bhagavat Memorial Mahajana Education Centre.
Affiliated to University of Mysore.

M.Sc. in Computer Science Regulations 2022-2023

Preamble

Mahajana Post Graduate Centre is an exclusive PG wing of SBRR Mahajana First Grade College (Autonomous). The centre happens to be the largest PG Centre affiliated to University of Mysore.

It was established in July 2003 with the motto "Enter to Learn, Depart to Serve". The Centre is affiliated to University of Mysore and offers Post Graduation programmes in the areas of direct relevance and value to the current generation of students. The Centre offers Post Graduate degree in 12 disciplines and is poised to start new programmes in the years to come.

M.Sc. in Computer Science was started in the year 2008. The programme is approved by University Grants Commission and affiliated to the University of Mysore. It is a four semester full time programme.

1. Definitions

Course

Every course offered will have three components associated with the teaching-learning process of the course, namely
(i) Lecture – L (ii) Tutorial- T (iii) Practical - P, where

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

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

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

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

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

If a course is of 4 credits then the different credit distribution patterns in L: T: P format could be
4 : 0 : 0, 1 : 2 : 1, 1 : 1 : 2, 1 : 0 : 3, 1 : 3 : 0,
2 : 1 : 1, 2 : 2 : 0, 2 : 0 : 2, 3 : 1 : 0, 3 : 0 : 1,
0 : 2 : 2, 0 : 4 : 0, 0 : 0 : 4, 0 : 1 : 3, 0 : 3 : 1,

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

Different courses of study are labelled and defined as follows:

Core Course

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

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

Elective Course

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

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

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

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

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

2. Eligibility for Admission

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

The candidates who have passed B.Sc with Computer Science / Computer Applications / Vocational Computer Applications / Computer Maintenance / Computer Systems as an optional course / BCA with minimum 45% marks in Cognate subject are eligible (relaxed to 40% in case of SC, ST and Category I Candidates). The candidates should have also studied Mathematics as a major or a minor subject in their B.Sc. / BCA degree.

3. Scheme of Instructions

3.1 A Masters Degree program is of 4 semesters-two year's duration for regular candidates. A regular candidate can avail a maximum of 8 semesters – 4 years (in one stretch) to complete Masters Degree (including blank semesters, if any). Whenever a candidate opts for blank semester(s)/DROP in a course or in courses or is compelled to DROP a course or courses as per the provision of the regulation, he/she has to study the prevailing courses offered by the department as per the prevailing scheme, when he/she continues his/her study.

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

Course Type	**Credits**
Hard Core	A minimum of 42, but not exceeding 52
Soft Core	A minimum of 16
Open Elective	A minimum of 4

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

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

3.4 Only such candidates who register for a minimum of 18 credits per semester in the first two semesters and complete successfully 76 credits in total of the 4 semesters be considered for declaration of ranks, medals and are eligible to apply for student fellowship, scholarship, free ships and hostel facilities.

3.5 In excess to the minimum of 76 credits for masters degree in the concerned discipline / subject of study, a candidate can opt to complete a minimum of 18 extra credits to acquire **add on proficiency diploma** in that particular discipline /subject along with the masters degree. In such of the cases where in, a candidate opts to earn at least 4 extra credits in different discipline / subjects in addition to a minimum of 76 credits at masters level as said above then an **add on proficiency certification** will be issued to the candidate by listing the courses studied and grades earned.

3.6 A candidate admitted to Masters Program can exercise an option to exit with Bachelor Honors Degree / PG diploma after earning 40 credits successfully.

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

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

4.1 Assessment and evaluation processes happen in a continuous mode. However, for reporting purposes, a semester is divided into 3 discrete components identified as C1, C2, and C3.

4.2 The performance of a candidate in a course will be assessed for a maximum of 100 marks as explained below:

4.2.1 The first component (C1), of assessment is for 15 marks. This will be based on test/ assignment/seminar/quiz/group discussions. During the first half of the semester, the first 50% of the syllabus will be completed. This shall be consolidated during the 8th week of the semester. Beyond 8th week, making changes in C1 is not permitted.

4.2.2 The second component (C2), of assessment is for 15 marks. This will be based on test/ assignment/seminar/quiz/group discussions. The continuous assessment and scores of second half of the semester will be consolidated during the 16th week of the semester. During the second half of the semester the remaining units in the course will be completed.

4.2.3 The outline for continuous assessment activities for Component-I (C1) and Component-II (C2) will be proposed by the teacher(s) concerned before the commencement of the semester and will be discussed and decided in the respective Departmental Council. The students should be informed about the modalities well in advance. The evaluated courses/assignments during component I (C1) and component II (C2) of assessment are immediately returned to the candidates after obtaining acknowledgement in the register maintained by the concern teacher for this purpose.

4.2.4 During the 18th -20th week of the semester, a semester-end examination of 3 hours duration shall be conducted for each course. This forms the third/final component of assessment (C3) and the maximum marks for the final component will be 70.

4.2.5 In case of a course with only practical component a practical examination will be conducted with two examiners (one internal and one external).

A candidate will be assessed on the basis of:
a) Knowledge of relevant processes
b) Skills and operations involved
c) Results / products including calculation and reporting.
If external examiner does not turn up then both the examiners will be internal examiners. The duration for semester-end practical examination shall be decided by the departmental council.

4.2.6 **Scheme of Valuation for Practical Examination:**
The student is evaluated for 70 marks in C3 as per the following scheme:
There will be two questions. A candidate has to prepare procedure for both the questions and execute:

Procedure Development	:	10 x 2=20 Marks
Implementation	:	15 x 2=30 Marks

Viva	:	10 Marks
Record	:	10 Marks
Total	**:**	**70 Marks**

*For change of question, 10 Marks will be deducted per question.

4.2.7 If **X** is the marks scored by the candidate out of 70 in C3 in theory examination, if **Y** is the marks scored by the candidate out of 70 in C3 in Practical examination, and if **Z** is the marks scored by the candidate out of 70 in C3 for a course of (L=0):T:(P=0)type that is entirely tutorial based course, then the final marks (M) in C3 is decided as per the following table.

L.T.P distribution	Find mark M in C3
L:T:P	[(L+T)*X]+[(T+P)*Y] L+2T+P
L:(T=0):P	(L*X)+(P*Y) L+P
L:T:(P=0)	X
L:(T=0):(P=0)	X
(L=0):T :P	Y
(L=0): (T=0):P	Y
(L=0): T:(P=0)	Z

4.2.8 The details of continuous assessment are summarized in the following table:

Component	Syllabus in a course	Weightage	Period of Continuous assessment
C1	First 50%	15%	First half of the semester To be consolidated by 8th week
C2	Remaining 50%	15%	Second half of the semester. To be consolidated by 16th week
C3	Semester-end examination (All units of the course)	70%	To be completed during 18th - 20th Week.
Final grades to be announced latest by 24th week			

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

4.2.10 **Finally, awarding the grades should be completed latest by 24th week of the semester**.

4.3 Minor/ Major Project Evaluation

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

Component – I (C1): Periodic Progress and Progress Reports (15%)
Component – II (C2): Results of Work and Draft Report (15%)
Component– III (C3): Final Viva-voce and evaluation (70%).

The report evaluation is for 40% and Viva-voce examination is for 30%.

4.4 In case a candidate secures less than 30% in C1and C2 put together in a course, the candidate is said to have DROPPED that course, and such a candidate is not allowed to appear for C3 in that course. In case a candidate's class attendance in a course is less than 75%, the candidate is said to have DROPPED that course, and such a candidate is not allowed to appear for C3 in that course.

Teachers offering the courses will place the above details in the Department Council meeting during the last week of the semester, before the commencement of C3, and subsequently a notification pertaining to the above will be brought out by the Chairman of the Department before the commencement of C3 examination. A copy of this notification shall also be sent to the office of the Director & Controller of Examinations.

4.5 In case a candidate secures less than 30% in C3, he/she may choose DROP/MAKEUP option.
In case a candidate secures more than or equal to 30% in C3, but his/her grade(G) = 4, as per section 4.7 below, then he/she may be declared to have been conditionally successful in this course, provided that such a benefit of conditional clearance based on G=4 shall not be availed for more than 8 credits for the entire programme of Master's Degree of two years.

In case a candidate secures less than 30% in C3, he/she may choose DROP/MAKE-UP option. The candidate has to exercise his/her option to DROP/MAKE UP immediately within 10 days from the date of notification of results.

A MAKE UP examination for C3 shall be conducted in all the semesters.
Candidates can register for the MAKE UP examination within 10 days from the date of notification of results. The MAKE UP examination will be conducted within one month of the notification of the results.

If a candidate is still unsuccessful, A MAKE UP Examination for odd semester courses will be conducted along with next regular odd semester examinations and for even semester courses along with next regular even semester examinations; however, not exceeding double the duration norm in one stretch from the date of joining the course.

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

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

Marks(M)	**Grade**	**Grade Point (GP = V x G)**
30-39	4	V*4
40-49	5	V*5
50-59	6	V*6
60-64	6.5	V*6.5
65-69	7	V*7
70-74	7.5	V*7.5
75-79	8	V*8
80-84	8.5	V*8.5

85-89	9	V*9
90-94	9.5	V*9.5
95-100	10	V*10

Here, **P** is the percentage of marks (P = [(C1+C2)+M] secured by a candidate in a course which is rounded to nearest integer. **V** is the credit value of course. **G** is the grade and GP is the grade point.

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

4.9 Overall Cumulative Grade Point Average (CGPA) of a candidate after successful Completion the required number of credits (76) is given by:

CGPA = ΣGP / Total number of credits

5. Classification of Results

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

CGPA	**FGP**	
	Numerical Index	**Qualitative Index**
4 <= CGPA < 5	5	Second Class
5 <= CGPA < 6	6	
6 <= CGPA < 7	7	First Class
7 <= CGPA < 8	8	
8 <= CGPA < 9	9	Distinction
9 <= CGPA < 10	10	

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

6. Medium of Instruction

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

7. Provision for Appeal

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

2. One senior faculty member (other than those concerned with the evaluation of the course concerned) drawn from the department/discipline and/or from the sister departments/sister disciplines.
3. One senior faculty member / course expert drawn from outside the department.

8. Any other issue not envisaged above, shall be resolved by the competent authority of the autonomous college, which shall be final and binding.

9. Any matter which is not covered under this regulation shall be resolved as per the College/ University of Mysore regulations

Programme Outcomes – M.Sc. Computer Science

PO 1: Apply the theoretical knowledge of Mathematics to design and develop models to solve real-time problems.

PO 2: Apply skills learnt in emerging technologies to construct and implement software systems of varying complexities.

PO 3: Communicate and engage effectively with diverse systems, processes and people to construct computer based solutions to problems.

PO 4: Recognize the need for and develop effective communication skills to engage in continuing professional development.

PO 5: Demonstrate the understanding of the concepts learnt relating to professional, ethical, legal, and social issues and responsibilities in real-life.

PO 6: Develop strong programming skills to implement research projects

SBRR MAHAJANA FIRST GRADE COLLEGE (Autonomous)

POST GRADUATE WING

(Accredited by NAAC with 'A' grade)

Pooja Bhagavat Memorial Mahajana Education Centre.
Affiliated to University of Mysore.

Scheme and Syllabi for M.Sc. in Computer Science

w.e.f. 2022-23

I Semester

Sl. No	Course Title	Type	Credits			Corse Code
			L	T	P	
1	Discrete Mathematical Structures	HC	3	1	0	22J1H1
2	Advanced Data Structures	HC	3	1	0	22J1H2
3	Database Technologies	HC	3	0	1	22J1H3
Soft Core Courses (Choose at most 2 Courses)						
1	Java Programming	SC	3	0	1	22J1S1
2	Operating Systems	SC	3	1	0	22J1S2
3	Web Technologies	SC	2	1	1	22J1S3
4	Computer Graphics	SC	2	1	1	22J1S4
5	Computer Architecture	SC	4	0	0	22J1S5
6	Numerical Algorithms	SC	4	0	0	22J1S6

II Semester

Sl. No	Course Title	Type	Credits			Corse Code
			L	T	P	
1	Design And Analysis of Algorithms	HC	2	1	1	22J2H1
2	Python Programming	HC	3	0	1	22J2H2
3	Data Communication & Networks	HC	3	1	0	22J2H3
Soft Core Courses (Choose at most 2 Courses)						
1	System Software	SC	3	0	1	22J2S1
2	Communication Skills	SC	4	0	0	22J2S2
3	Professional Ethics and Values	SC	3	1	0	22J2S3
4	Pattern Recognition	SC	3	1	0	22J2S4

5	Big Data Analytics	SC	3	0	1	22J2S5
	World Wide Web	OE	3	1	0	22J2E1

III Semester

Sl. No	Course Title	Type	Credits			Corse Code
			L	T	P	
1	Theory of Languages	HC	3	1	0	22J3H1
2	Machine Learning	HC	3	0	1	22J3H2
3	Minor Project	HC	0	1	3	22J3H3
Soft Core Courses (Choose at most 2 Courses)						
1	Artificial Intelligence	SC	3	1	0	22J3S1
2	Digital Image Processing	SC	3	0	1	22J3S2
3	C# Programming	SC	3	0	1	22J3S3
4	Android Programming	SC	3	0	1	22J3S4
5	Software Engineering	SC	3	1	0	22J3S5
	E-Commerce	OE	3	1	0	22J3E1

IV Semester

Sl. No	Course Title	Type	Credits			Corse Code
			L	T	P	
1	Dissertation	HC	0	2	10	22J4H1
Soft Core Courses (Choose at most 2 Courses)						
1	Compiler Construction	SC	3	1	0	22J4S1
2	Advanced Database Management System	SC	2	1	1	22J4S2
3	Data Mining	SC	3	0	1	22J4S3
	Office Automation	OE	3	1	0	22J4E1

HC DISCRETE MATHEMATICAL STRUCTURES [3:1:0]

Objectives:

- Learn the fundamentals of counting theory, set theory, logic, quantifiers, and relations.
- Learn different proof techniques like direct or indirect, proof by contradiction, check the validity of a given argument.
- Understand the concepts of functions and relations to solve a given problem.
- Learn the concepts of graph theory and applications.

Outcomes:

- Apply the concepts of set theory, logic, quantifiers and relations in specifying and solving problems.
- Identify the quantifiers and their uses and Make use of fundamentals of logic theory.
- Apply the mathematical induction principle and different methods to solve the given problem.
- Make use of basic concepts of graph theory to solve the given problem.

Unit I

Principles of Counting: The Rules of Sum and Product, Permutation, Combinations, combinations with repetition and Problems.

Sets and Subsets: Set Operations, Membership table method and Venn diagram method and the Laws of Set Theory, Addition principle-Counting and Venn Diagrams, A First Word on Probability.

Unit II

Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic theory, Logical Implication – Rules of Inference. Argument – Definition, validity and invalidity.

The Use of Quantifiers: Quantifiers, Definitions, Argument representation using quantifiers, validity.

Proofs of Theorems- Direct and Indirect method - contradiction and contra positive method.

Unit III

Relations and Functions: Properties of the Integers: Mathematical Induction, The Well Ordering Principle- Mathematical Induction (Alternative form) (problems), Recursive Definitions

Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions – Stirling Numbers of the Second Kind, Special Functions, The Pigeon-hole Principle, Function Composition and Inverse Functions.

Unit IV

An Introductions to Graph Theory: Definitions and examples Sub graphs, Complements, and Graph Isomorphism, Vertex Degree : Euler Trails and Circuits.

Graph coloring and Chromatic Numbers. Definitions, Properties and examples rooted trees, Trees and sorting. Weighted Trees and Prefix codes. Spanning trees- minimal spanning tree by Prim’s and Krushkal’s Algorithm.

References

1. Discrete and Combinatorial Mathematics, Ralph P. Grimaldi, 5th Edition, Pearson Education.
2. Discrete Mathematics and its Applications, Kenneth H. Rosen, 7th Edition, McGraw Hill.

3. Discrete Mathematical Structures with Applications to Computer Science by Tremblay and Manohar, McGraw-Hill Publications.
4. A Treatise on Discrete Mathematical Structures, Jayant Ganguly, Sanguine-Pearson.
5. Discrete Mathematical Structures –by Dr. D.S. Chandrashekhariah.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	1	2	-	-	1
CO 2	3	1	2	-	-	1
CO 3	3	2	2	-	-	2
CO 4	3	2	2	-	-	1
Weighted Average	3	1.5	2	0	0	1.25

1: Low, 2: Moderate, 3: High

HC ADVANCED DATA STRUCTURES [3:1:0]

Objectives:

- Learn about and understand different data structures like dictionaries, hash tables, priority queues, and different types of search trees.
- Understand how the above data structures can be represented.
- Understand how different operations like insertion, deletion, searching, etc. can be implemented in the above mentioned data structures.
- Understand string matching algorithms and operations on tries.

Outcomes:

- Understand the ADT specification of dictionary data structure, priority queue and binary search trees.
- Perform insertion, deletion and searching operation on dictionary, priority queue and binary search trees.
- Perform the sorting using external sorting.
- Identify the applications of string matching algorithms and tries.

Unit I

Dictionaries, linear list representation, skip list representation, operations insertion, deletion and searching, hash table representation, hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, comparison of hashing and skip lists.

Unit II

Priority Queues — Definition, ADT, Realizing a Priority Queue using Heaps, Definition, insertion, Deletion, External Sorting- Model for external sorting, Multi-way merge, Poly-phase merge.

Unit III

Search Trees , Binary Search Trees, Definition, ADT, Implementation, Operations- Searching, Insertion and Deletion, AVL Trees, Definition, Height of an AVL Tree, Operations — Insertion, Deletion and Searching, Introduction to Red —Black and Splay Trees, B-Trees, B-Tree of order m, Comparison of Search Trees

Unit IV

Pattern matching and Tries: Pattern matching algorithms-Brute force, the Boyer —Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries and Suffix tries.

References

1. Data structures, Algorithms and Applications in C++, S.Sahni, University Press (India), 2nd edition, Universities Press Orient Longman.
2. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and Mount, Wiley student edition, John Wiley and Sons.
3. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education, Second Edition.
4. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Thomson.
5. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	1	2	-	1	1
CO 2	1	2	2	-	2	1
CO 3	3	2	2	-	1	2
CO 4	2	2	1	-	1	1
Weighted Average	2.25	1.75	1.75	-	1.25	1.25

1: Low, 2: Moderate, 3: High

HC DATABASE TECHNOLOGIES [3:0:1]

Objectives:

- Learn and practice data modelling using the entity-relationship and developing database designs.
- Understand the use of Structured Query Language (SQL) and learn SQL syntax.
- Apply normalization techniques to normalize the database.
- Comprehend the needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Outcomes:

- Comprehend data models and schemas in DBMS.
- Use SQL- the standard language of relational databases.
- Understand the functional dependencies and design of the database.
- Understand the concept of Transaction and Query processing.

Unit I

Overview of Database Systems and Entity- Relationship Model

A historical perspective, file system versus a DBMS, advantages of a DBMS, levels of abstraction in a DBMS, structure of a DBMS, users of databases, entity, entity types, entity sets, attributes, keys, relationships, relationship sets and additional features of ER-model-key constraints, participation constraints and weak entities.

Unit II

Relational model, Relational Algebra and Structured Query Language

Relational model- Concepts, relational constraints and relational database schemas. Relational algebra - Basic and additional relational operations with examples. Data definition, constraints and schema changes in SQL, Basic queries in SQL: insert, delete and update statements and joins in SQL, views in SQL.

Unit III

Database Design, Overview of storage and indexing

Informal design guidelines for relational schemas, functional dependencies, normal forms, general definitions of first, second, third and boyce-codd normal forms.

File organization and indexing: sequential file organization, heap file organization, clustered indexes primary and secondary indexes, hash based indexing and B+ tree-based indexing.

Unit IV

Overview of transaction management

The ACID properties, consistency and isolation, atomicity and durability, transaction on schedules, concurrent execution of transactions, motivation for concurrent execution, serializability, anomalies due to interleaved execution, lock-based concurrency control, strict two phase locking and performance of locking.

References

1. Fundamentals of Database Systems by Navathe and Elmasri –Pearson Education, Fifth Edition.
2. Database Systems Concepts, 3rd edition by Abraham Silberschatz, Henry Korth and S. Sudarshan, Tata McGraw Hill.
3. Principles of database systems by Ullman, Computer Science press.

4. DBMS by Prof. S.Nandagopalan, 7th Revised Edition.

Course articulation matrix:

PO / CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	1	2	2	-	1	1
CO 2	2	3	3	-	2	2
CO 3	1	2	1	-	1	1
CO 4	1	2	1	-	1	-
Weighted Average	1.25	2.25	1.75	-	1.25	1

1: Low, 2: Moderate, 3: High

SC JAVA PROGRAMMING [3:0:1]

Objectives:

- To gain an understanding of the object oriented paradigm and concepts in general.
- Learn the basics of the Java environment and the programming language.
- To learn how to implement different object oriented concepts in Java.
- To learn about using files and creating packages.

Outcomes:

- Understand different aspects of object oriented paradigm and programming fundamentals.
- Build programs using programming basics, class fundamentals and reusable code using inheritance and polymorphism.
- Model solutions using files and interfaces.
- Develop efficient and error free applications using packages and exceptions..

Unit 1: Object-Oriented Programming (OOPS) concepts:

Classes & Objects, Pillars Of Object Oriented Programming, OOPS concepts and terminology, Encapsulation & Examples, Abstraction & Examples, Inheritance: Advantages of OOPS, About Java, Execution Model Of Java, Bytecode, First Java Program, Compiling and Interpreting Programs, The JDK Directory Structure, Data types and Variables: Primitive & non-Primitive Datatypes & Declarations, Variables & Types, Numeric & Character Literals, String formatting and Parsing, String Literals, The Dot Operator.

Unit 2: Methods:

Methods: Method Structure, Declaration Of Methods, Calling Of Methods, Defining Methods, Method Parameters Scope, static methods, Operators and Expressions:

Expressions, Operator Precedence, The Cast Operator, Control Flow Statements, While and do-while Loops, for Loops, The continue Statement, The break Statement, Objects and Classes: Defining a Class, Creating an Object, Accessing Class Members, Instance Data and Class Data, Defining Methods, Constructors, Access Modifiers, Inheritance & Polymorphism: Inheritance in Java, Types Of Inheritance, Method Overloading, Run-time Polymorphism, Method Overriding, super keyword.

Unit 3 : Java Files and I/O:

Streams, Reading and Writing to Files (only txt files), Input and Output Stream classes, using the file class, Using Streams, creation of files, reading/writing characters, bytes, Interfaces and Abstract Classes: Interface: Defining Interfaces, Separating Interface and Implementation, Implementing and Extending Interfaces, Abstract Classes.

Unit 4 : Packages:

Package, Advantages of using a Package, Types Of Packages, Naming Convention, Steps For Creating Packages, The import Statement, Static Imports, CLASSPATH and Import, Defining Packages, Package Scope, Exception Handling: Exceptions Overview, Exception Keywords, Catching Exceptions, The finally Block, Exception Methods, Declaring Exceptions, Defining and Throwing Exceptions, Errors and Runtime Exceptions, Assertions.

References

1. Programming with JAVA- A Primer, E. Balagurusamy, Tata Mc-Graw-Hill.
2. JAVA for you- P Koparkar, Tata Mc-Graw-Hill.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	2	-	2	3
CO 2	-	3	2	-	2	3
CO 3	-	3	3	-	2	3
CO 4	-	3	3	-	2	3
Weighted Average	0	3	2.5	0	2	3

1: Low, 2: Moderate, 3: High

SC OPERATING SYSTEMS [3:1:0]

Objectives:

- To learn about the fundamental principles of operating system, processes and their communication
- To learn about various operating system issues related to process management management like threads, process scheduling, synchronisation and deadlocks.
- To learn about various memory management techniques, including virtual memory, paging and segmentation.
- To know about disk and file management and the distributed file system concepts.

Outcomes:

- Able to comprehend the operating system components and its services
- Able to understand how process is created and various process related components of the operating system.
- Able to comprehend how memory management and virtual memory management is done.
- Able to understand different file and directory structures and how files are stored in secondary storage.

Unit I

Introduction -Computer System Organisation – Computer system architecture – Operating system operations - Operating systems services-System calls- Types of system calls – Operating system structure.

Processes-process concept- process scheduling-operation on processes. Threads – Multithreading models – Threading issues.

Unit II

Process Scheduling - Scheduling criteria-Scheduling algorithms – Thread scheduling - Multiple-processor Scheduling.

Process Synchronization – Critical Section problem – Peterson's solution - Semaphores-Classical problems of synchronization - critical regions – Introduction to Montors.

Unit III

Deadlocks – System model - Deadlock Characterization - Deadlock handling - Deadlock Prevention - Deadlock avoidance - Deadlock Detection - Deadlock Recovery.

Memory Management – Swapping - Contiguous Memory allocation -Segmentation Paging.

Virtual Memory Management - Demand paging – Copy on write - Page Replacement - Thrashing.

Unit IV

File System – File concept – Access methods – Directory structure – Directory and disk structure - File Systems structures - Directory Implementation - Allocation Methods - Free Space management.

Case Study : Linux System – Linux history, Design Principles, Kernel modules.

References

1. "Operating Systems Concepts", Abraham Silberschalz Peter B Galvin, G.Gagne, 9th Edition, John Wiley & Sons.
2. "Modern operating Systems", Andrew S.Tanenbaum, Third Edition, PHI.
3. "Operating Systems: A Concept-based Approach", D M Dhamdhere, Second Edition, Tata McGraw-Hill.

4. “Operating Systems”, H M Deital, P J Deital and D R Choffnes, 3rd edition, Pearson Education.
5. “Operating Systems: Internals and Design Principles”, William Stallings, Seventh Edition, Prentice Hall.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	1	-	-	1
CO 2	-	3	1	-	-	1
CO 3	-	3	1	-	-	1
CO 4	-	3	1	-	-	1
Weighted Average	-	3	1	-	-	1

1: Low, 2: Moderate, 3: High

SC WEB TECHNOLOGIES [2:1:1]

Objectives:

- To help students understand the basis of Internet and also the basic building blocks of web pages using HTML and CSS.
- To help students understand and use Java script and the Document Object Model.
- To help students understand and use PHP for back-end, especially database connectivity.
- To help students understand the use of web frameworks and content management systems for creating and managing websites faster and easier.

Outcomes:

- Develop an ability to implement HTML5 pages using fundamental tags.
- Develop style sheet using CSS for a given problem.
- Illustrate JavaScript to validate a form with event handler for a given problem.
- Determine PHP in the back-end for database connectivity, web frameworks and content management systems.

Unit I

Introduction to Internet, WWW, Web Browsers, and Web Servers, URLs, MIME, HTTP, Security. Quick introduction to HTML5 : basic text formatting, presentation elements, phrase elements, lists, Tables – attributes, grouping elements, basic links, email link, Image, Audio, Video, image maps , Forms.

Unit II

Cascading Style Sheet : Introduction, Levels of Style Sheet and specification formats, embedded style sheet, External Style Sheet, inline Style Sheet, Box Model, selector forms, Class and ID method, DIV and SPAN tags, Inheritance with CSS. Responsive Web Design.

Unit III

JavaScript: JavaScript in HTML, Language Basics – Variables, operators, statements, functions, Data type conversions, reference types, Document object Model : methods, HTML DOM Elements, changing HTML and CSS, Events and event handling, event listener, form validation. Browser Object Model : Window, screen, history, popup alert, timing, cookies.

Unit IV

PHP: Forms PHP, Form Handling, PHP MySQL connectivity.

Brief introduction to Web Frameworks and Content Management System, creation of a simple website using WordPress.

References

1. Internet and World Wide Web: How to Program - Paul Deitel, Harvey Deitel, Abbey Deitel, 5th Edition - 2018, Pearson Education.
2. HTML & CSS: The Complete Reference - Thomas Powell, 5th Edition – 2015, McGraw Hill Education.
3. HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery) - DT Editorial Services, 2nd Edition – 2016, Dreamtech Press.
4. Programming PHP: Creating Dynamic Web Pages - Kevin Tatroe, Peter MacIntyre, Rasmus Lerdorf, 3rd Edition – 2013, O’Reilly.
5. Learning PHP, MySQL & JavaScript with j Query, CSS & HTML5 - Robin Nixon, 4th Edition – 2015, O’Reilly.
6. https://www.w3schools.com/html/

7. https://www.w3schools.com/css/
8. https://www.w3schools.com/js/
9. https://www.w3schools.com/php/
10. https://en.wikipedia.org/wiki/Web_framework
11. https://en.wikipedia.org/wiki/Content_management_system

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	-	-	1	1
CO 2	-	3	-	-	1	1
CO 3	1	3	2	-	1	1
CO 4	1	3	2	-	1	1
Weighted Average	1	3	2	0	1	1

1: Low, 2: Moderate, 3: High

SC COMPUTER GRAPHICS [2:1:1]

Objectives:

- To provide an overview of various device level algorithms.
- To provide an understanding of homogeneous coordinates and various 2D and 3D transformations
- To provide an introduction to 3D concepts like projections, curves.
- To make the students understand how to implement the computer graphics concepts using OpenGL.

Outcomes:

- Able to identify and use various graphics hardware, basic coordinate representations, functions and scan conversion algorithms.
- Able to implement various filled area primitives, 2D transformations and viewing
- Able to implement 2D clipping algorithms 3D geometric transformations.
- Able to implement 3D viewing, spline curves and visible surface detection

Unit I

Graphics hardware: Video display devices, Raster-scan systems, Graphics software : Coordinate representations, Graphics functions, standards, Introduction to OpenGL.

Graphics Output Primitives: Coordinate reference frames, Two-Dimensional reference frame in OpenGL, OpenGL Point Functions, Line Functions, Curve functions.

Scan-Conversion: Line-Drawing Algorithms: DDA, Bresenhams, Setting frame-buffer values, Circle-Generating algorithms : Midpoint Circle Algorithm.

Unit II

Filled area primitives: Scan-line polygon fill algorithm, Boundary fill algorithm, Flood fill algorithm, Inside-outside tests. Brief overview on Ant aliasing methods.

2D geometrical transformations: Basic two-dimensional geometric transformations, Homogeneous Coordinates and Matrix Representation, Inverse Transformations, Brief overview of Composite transformations, Reflection, Shear, OpenGL functions for two-dimensional geometric transformations, Programming examples.

2D viewing: Windows and viewports, Two-dimensional viewing pipeline, clipping window, Normalization and viewport transformations, Brief overview of OpenGL 2D viewing functions.

Unit III

2D Clipping Algorithms: Point clipping, Line clipping: Cohen- Sutherland and Liang-Barsky Line clipping, polygon fill-area clipping: Sutherland-Hodgman algorithm, Text clipping.

3D geometrical transformations: 3D translation, 3D scaling. 3D rotation: coordinate-axis rotations, general 3D rotations, Other 3D transformations, Affine transformations, OpenGL geometric transformation functions.

Unit IV

Three-dimensional viewing: Overview, Three-dimensional viewing pipeline, Projection transformations, 3D viewing functions.

Spline representations : Interpolation and Approximation splines, parametric and Geometric continuity conditions, Bezier spline curves, B-Spline curves.

Visible surface detection : Classification of visible surface detection algorithms, Back- Face detection, Depth buffer method.

References

1. Computer Graphics with OpenGL, Donald D. Hearn, M. Pauline Baker, Warren Carithers, Fourth Edition, Pearson India Education Services.
2. Computer Graphics Principles & Practice in C, Foley, Vandam, Feiner, Hughes, Pearson Education.
3. Open GL Super Bible : Comprehensive Tutorial and Reference, Richard S Wright and Jr. Michael Sweet, 7th Edition, Pearson Education.
4. Computer Graphics, Roy A. Plastock, Gordon Kalley, Schaum's Outlines, McGraw Hill.
5. Computer Graphics 2nd Edition (Paperback) by Steven Harrington, Tata McGraw Hill.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	3	-	-	-	1
CO 2	3	3	-	-	-	1
CO 3	3	3	-	-	-	1
CO 4	3	3	-	-	-	1
Weighted Average	3	3	-	-	-	1

1: Low, 2: Moderate, 3: High

SC COMPUTER ARCHITECTURE [4:0:0]

Objectives:

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

Outcomes:

- Develop an ability to understand the concept of cache mapping techniques.
- Develop an ability to understand basics of organizational and architectural issues of a digital computer.
- Acquire knowledge and understanding the theory of Digital Design and Computer Organization to provide an insight to basic computer components.
- Develop an ability to conceptualize instruction level parallelism.

Unit I : Parallel Computer Models

The state of Computing - Evolution of Computer Architecture, System Attributes to Performance. Multiprocessors and Multicomputer Shared Memory Multiprocessors, Distributed – Memory Multicomputer Multivector and SIMD Computers – Vector Supercomputers, SIMD supercomputers conditions of Parallelism – Data and Resource Dependencies, Hardware and software parallelism.

Unit II : Processor and Memory Hierarchy

Process Technology – Instruction Pipelines, Processors and Coprocessors, Instruction Set Architectures, Representative CISC Processors, Representative RISC Processors, Superscalar Processors. Memory Technology, Inclusion, Coherence and Locality. Cache Memory organization – Cache Addressing modes, Direct mapping and Associative caches, Set Associative Cache. Shared – Memory organizations – Interleaved Memory organization

Unit III : Pipelining

Linear Pipeline processors – Asynchronous and Synchronous Models, Instruction Pipeline Design – Mechanisms for instruction Pipelining, Arithmetic Pipeline Design – Computer Arithmetic Principles, Arithmetic Pipeline Stage, Multifunctional Arithmetic Pipelines

Unit IV : Multiprocessors

Multiprocessor system Interconnects – Hierarchical Bus system, Cache Coherence Problem. Message – Passing Mechanisms – Message – Routing Schemes, Deadlock and Virtual Channels, Multithreaded Architecture – Multithreading Principles, Issues and Solutions

References

1. Advanced Computer Architecture – Kai Hwang – Tata McGraw Hill.
2. Parallel Computer Architecture, David E Culler, J.P.Singh and Anoop Guptha.
3. Computer Architecture and Organization – John. P. Hayes – Third Edition –Tata McGraw Hill.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	1	1	1	-	1	-
CO 2	1	1	1	-	-	-
CO 3	1	1	1	-	-	-
CO 4	1	1	1	-	-	-
Weighted Average	1	1	1	0	1	0

1: Low, 2: Moderate, 3: High

SC NUMERICAL ALGORITHMS [4:0:0]

Objectives:

- To introduce the different types of errors in computing
- Finding the roots of the non-linear equations, Numerical integration and Ordinary differential equations.
- Finding solutions of simultaneous linear algebraic equations.
- Introducing interpolation and statistical methods .

Outcomes:

- Demonstrate understanding of common numerical methods and how they are used to obtain approximate solutions.
- Apply numerical methods to obtain approximate solutions to mathematical problems.
- Derive numerical methods for various mathematical operations and tasks such as solution of non-linear equations, numerical integration and ordinary differential equations.
- Gain an understanding of interpolation and statistical methods.

Unit I

Introduction to Numerical Computing: Introduction, Numeric Data, Analog Computing, Digital Computing, Process of Numerical Computing and Characteristics of Numerical Computing.

Approximations and Error in Computing: Introduction, Significant Digits, Inherent Errors, Numerical Errors, Modelling errors, Blunders, Absolute and relative Errors, Blunders and Error Propagation.

Roots of Nonlinear Equations: Bisection method, False position method, Newton Raphson method and Secant method.

Unit II

Numerical Integration: Trapezoidal rule, Simpson's 1/3rd rule and Simpson's 3/8th rule.

Ordinary Differential Equations: Euler's method, Modified Euler's method, Runge-Kutta II and IV order methods.

Unit III

Solutions of Simultaneous Linear Algebraic Equations: Gauss Elimination method, Gauss Jordan method and LU Decomposition method.

Iterative methods: Jacobi's iterative method and Gauss-Seidel iterative method.

Unit IV

Interpolation: Newton-Gregory forward interpolation, Newton-Gregory backward interpolation, divided differences, Newton's divided difference and Lagrange's interpolation.

Statistical methods: Introduction, Definitions, Classifications, Frequency Distribution, Mean – Arithmetic Mean for grouped and ungrouped data and Geometric Mean for grouped and ungrouped data.

References

1. Numerical Methods – E Balaguruswamy, Tata McGraw-Hill.
2. Engineering Mathematics Vol. III - A by Dr. K.S. Chandrashekar, Sudha Publcations.
3. Computer Oriented Numerical Methods by Rajaraman V.
4. Fundamentals of Mathematical Statistics by Gupta and Kapoor

5. Probability and Statistics for engineers and scientists by Ronald E. Walpole and Raymond H Mayers
6. Mathematical Statistics by John Freund.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	**3**	-	-	-	-	-
CO 2	**3**	-	-	-	-	-
CO 3	**3**	-	-	-	-	-
CO 4	**3**	-	-	-	-	-
Weighted Average	**3**	-	-	-	-	-

1: Low, 2: Moderate, 3: High

HC DESIGN AND ANALYSIS OF ALGORITHMS [2:1:1]

Objectives:

- Comprehend the performance analysis of an algorithm.
- Understand time and space complexity of various data structures.
- Comprehend time and space complexities of an algorithm.
- Learn different design strategies like divide and conquer, transfer and conquer, greedy, dynamic programming, backtracking and branch and bound

Outcomes:

- Compare between different data structures. Pick an appropriate data structure for a design situation. Analyze Performance of algorithms using asymptotic analysis.
- Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.
- Describe the greedy paradigm and dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize greedy algorithms, and analyze them.
- Describe the backtracking paradigm and branch and bound paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize greedy algorithms, and analyze them.

Unit I : Introduction

Algorithms, structured algorithms, analysis of algorithms, complexity analysis and profiling, asymptotic complexity, review of stack, queues, Recursion, heaps and heap sort, case studies(complexity analysis and profiling)- prime and Fibonacci numbers, GCD and LCM, sorting algorithms- selection sort, bubble sort and insertion sort.

Unit II : Divide and conquer & Transfer and conquer

Divide and conquer general method, binary search, Maximum and minimum element in list, merger sort, quick sort. Transfer and Conquer – solution to simultaneous equations by triangularization, diagonalization algorithms.

Unit III : Greedy method and Dynamic programming

Greedy method-General method, optimal storage on tapes, knapsack problem, optimal merge pattern, Minimum cost spanning trees (prim's algorithm and Kruskal’s algorithm), single source shortest paths. Dynamic Programming-General methods, multistage graphs, all pair’s shortest paths, Travelling salesman problem, 0/1 Knapsack problem

Unit IV : Backtracking and Branch and Bound

General method for backtracking, 8-queen Problem, sum of subsets problem, Graph Colouring problem.

Branch and Bound general method, 0/1 knapsack problem, travelling salesman problem.

References

1. “Fundamentals of Computer Algorithms” Ellis Horowitz, Sartaj Sahni and Sanguthevar, Rajasekaran Galgotia Publications.
2. ”Introduction to the Design & Analysis of Algorithms”, Anany V. Levitin Pearson Education, 3rd edition.

3. "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	-	1	-	1	-
CO 2	3	-	-	-	2	-
CO 3	3	-	-	-	2	-
CO 4	3	-	-	-	2	-
Weighted Average	3	0	1	0	1.75	

1: Low, 2: Moderate, 3: High

HC PYTHON PROGRAMMING [3:0:1]

Objectives:

- Understand programming paradigms brought in by Python.
- To learn to use python for text processing, with a focus on Regular Expressions, List and Dictionaries.
- To explore various modules and libraries to cover the landscape of Python programming.

Outcomes:

- Demonstrate the use of the built -in objects of Python
- Demonstrate significant experience with the Python program development environment.
- Understand and implement the basic methods of python modules like NumPy, and Pandas.
- Visualize data using Matplotlib module.

Unit-1 Introduction to Python

Structure of Python Program, Branching and Looping, Numbers, Strings, Lists, tuples, functions, built-in methods for strings, lists, tuples. List comprehensions.

Unit-2 Sequence Datatypes and Object-Oriented Programming

Sets, Dictionaries, Classes: Classes and Instances, Inheritance, Exceptional Handling, Introduction to Regular Expressions using “re” module.

Unit-3 Using NumPy and Pandas

Basics of NumPy: Computation on NumPy, Aggregations, Computation on Arrays, Comparisons, Masks and Boolean Arrays, Fancy Indexing, Sorting Arrays, Structured Data: NumPy’s Structured Array.

Introduction to Pandas Objects: Data indexing and Selection, Operating on Data in Pandas, Handling Missing Data, Hierarchical Indexing, Combining Data Sets

Unit-4 Data Visualization with Matplotlib

Basic functions of Matplotlib: Simple Line Plot, Scatter Plot, Density and Contour Plots, Histograms, Binnings and Density, Customizing Plot Legends, Colour Bars, Three-Dimensional Plotting in Matplotlib.

References:

1. The Python Tutorial : https://docs.python.org/3/tutorial/index.html
2. Python Data Science Handbook - Essential Tools for Working with Data, Jake VanderPlas , O’Reily Media,Inc, 2016
3. An Introduction to Python and Computer Programming, Zhang.Y, Springer Publications,2016
4. NumPy : https://numpy.org/
5. Pandas : https://pandas.pydata.org/
6. Matplotlib : https://matplotlib.org/
7. Core Python Applications Programming, 3rd Edition by Wesley J. Chun
8. Python, The complete Reference, Martin C. Brown, McGraw Hill Education.
9. Python in a Nutshell, A. Martelli, A. Ravenscroft, S. Holden, OREILLY.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	2	-	-	2	3
CO 2	3	3	-	-	3	3
CO 3	-	3	-	-	2	1
CO 4	2	2	-	-	2	3
Weighted Average	2	2.5	-	-	2.25	2.5

1: Low, 2: Moderate, 3: High

HC DATA COMMUNICATION & NETWORKS [3:1:0]

Objectives:

- Understand the basics of data communication components.
- Learn the protocols of Data link layer.
- Understand different network layer services and routing protocols
- Know the different techniques involved transport layer and application layer

Outcomes:

- Understand and implement various types of transmissions in wired and wireless communications
- Study and develop the aspects of communication channels of Data Link Layer.
- Understand Design & apply various routing protocols of the Networks Layer.
- Design applications using the protocols of Transport & application Layer.

Unit I: Data Communications

Components, Direction of Data Flow, Networks –Network Criteria and Network Types, TCP/IP Protocol suite, OSI Model, Multiplexing, Transmission media-Guided and Unguided media.

Unit II: Data link layer

Introduction, Data-Link Control-Framing, Error control, Protocols-Noiseless Channels and Noisy Channels, Medium Access Sub Layer-ALOHA, CSMA/CD, Wired LAN – Ethernet, Wireless LAN – IEEE 802.11

Unit III: Network layer

Network Layer: Internet Protocol – IPv4, Ipv6, IPv4 addresses, IPv6 addresses, Transition from IPv4 to IPv6, Routing algorithms, Unicast Routing protocols-Internet Structure, Brief intoduction to RIP, OSPF and BGP, Unicasting vs. Multicasting.

Unit IV: Transport Layer and Application Layer

Transport layer services- Process to process communication, Addressing, Transport layer protocols-Services, Port numbers, UDP and TCP, Application Layer: Client/Server Paradigm, Standard Applications : WWW and HTTP, FTP, Electronic Mail, TELNET, SSH, DNS, Introduction to P2P networks.

References:

1. Data Communications and Networking with TCPIP Protocol Suite - Behrouz A. Forouzan, 6th Edition, McGraw Hill.
2. Computer Networks - Andrew S Tanenbaum, 5th Edition. Pearson Education, PHI.
3. Data communications and Computer Networks - P.C .Gupta, PHI.
4. An Engineering Approach to Computer Networks - S. Keshav, 2nd Edition, Pearson Education.
5. Understanding communications and Networks, 3rd Edition, W.A. Shay, Cengage Learning.
6. Computer Networking: A Top-Down Approach Featuring the Internet - James F. Kurose & Keith W. Ross, 3rd Edition, Pearson Education.
7. Data and Computer Communication- William Stallings, 6th Edition, Pearson Education.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	3	-	2	2
CO 2	-	-	3	2	-	-
CO 3	-	3	3	3	-	-
CO 4	-	2	3	2	3	3
Weighted Average	-	2.66	3	2.33	2.5	2.5

1: Low, 2: Moderate, 3: High

SC SYSTEM SOFTWARE [3:0:1]

Objectives:

- To understand the design of an assembler for a simple machine architecture.
- To understand the need and design of a macro processing facility.
- To learn about loading, different loading schemes and issues related to it, and implementation of a loader.
- To get an overview of compiler functions and learn about basic lexical analysis and parsing.

Outcomes:

- Develop an Ability to master the design of assembler.
- Able to understand various issues related to processing macros.
- Able to understand different loaders schemes, and related issues.
- Develop ability to write simple lexical analyser and parser with Lex and Yacc.

Unit I

Introduction, general machine structure, general approach to a new machine, assemblers, general design procedure, design of assembler- statement of problem, data structure, format of data bases, algorithm, look for modularity.

Unit II

Macro language and the macro processor – macro instructions, features of a macro facility, macro instruction arguments, conditional macro expansion, macro calls within macros, macro instructions defining macros, implementation of a restricted facility.

Unit III

Loaders, Loader schemes, design of an absolute loader, design of a direct linking loader-specification of problem, specification of data structures, format of data bases, algorithm.

Unit IV

Introduction to Compilers : Language Processors, Structure of a Compiler.

Introduction to Lex and Yacc: The Simplest Lex Program, Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, Using LEX, Using YACC – Grammars, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions

References

1. "Systems Programming", John J. Donovan, Tata McGraw-Hill.
2. Compilers: Principles, Techniques, and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, 2nd Edition, Pearson.
3. Lex & Yacc, John R. Levine, Tony Mason, Doug Brown, 2nd Edition, O'Reilly.
4. System Software: An introduction to system programming, Leland L.Beck and D.Manjula, 3rd edition.
5. Systems Programming and Operating Systems, D. M. Dhamdhere, Second Revised Edition, Tata McGraw-Hill.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	1	2	2	-	-	3
CO 2	1	2	2	-	-	3
CO 3	1	3	2	-	-	3
CO 4	3	3	2	-	-	3
Weighted Average	1.5	2.5	2	0	0	3

1: Low, 2: Moderate, 3: High

SC COMMUNICATION SKILLS [4:0:0]

Objectives:

- The factors governing good communication and how good communication skills can be developed.
- How good communication skills are a critical building block to both personal and business success.
- How to use effective communication skills in business.
- The need to modify communication depending on business situation and circumstances.

Outcomes:

- Understand and apply knowledge of human communication and language processes as they occur across various contexts from multiple perspectives.
- Understand and evaluate key theoretical approaches used in the interdisciplinary field of communication.
- Find, use, and evaluate primary academic writing associated with the communication discipline.
- Communicate effectively orally and in writing.

Unit I

Importance of communication, its basic model, formal and informal communications, barriers to communication, 6 C's of Communication, feedback and its effectiveness, Non- Verbal communication - Etiquettes.

Unit II

Oral communication, Speaking: Paralanguage: Sounds, stress, intonation- Art of conversation – Presentation skills, – Public speaking- Expressing Techniques, importance of listening, role of visual aids, persuasive communication, Group Discussion.

Unit III

Written communication – Effective writing – Paragraph – Essay- Reports – Letters- Articles – Notices, Agenda & Minutes, Email Etiquetttes.

Unit IV

Interview sills: Types of Interviews – Preparing for interview – Preparing a CV – Structuring the interview- Mock Interview - Quick Tips.

References

1. Soft skills: know yourself & know the world, Dr. Alex K.
2. Communication for results – C Hamilton & Parker.
3. Instrument of Communication – P Meredith.
4. Basic Management skills for all – E H McGrath.
5. Managerial Communication – P M Timm.
6. Thesis and Assignment writing – Anderson.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	-	2	3	1	-
CO 2	1	-	3	-	2	2
CO 3	-	3	3	3	-	-
CO 4	1	2	3	3	2	3
Weighted Average	1	2.5	2.75	3	1.66	2.5

1: Low, 2: Moderate, 3: High

SC PROFESSIONAL ETHICS AND HUMAN VALUES [3:1:0]

Objectives:

- Understand the fundamentals of Human values.
- Know the concepts of engineering ethics and responsibilities.
- Learn about the Business Intelligence lifecycle and its methodologies.
- Get an overview of Global issues and its practices.

Outcomes:

- Implement the aspects of Human Values.
- Interpret the ethics of engineering and its associated responsibilities.
- Employ the code of ethics in their profession.
- Display the awareness of Global issues in Ethics .

Unit I: Human Values

Objectives, Morals, Values, Ethics, Integrity, Work ethics, Respect for others, living peacefully, Honesty, Courage, Valuing time, Cooperation, Commitment, Self-confidence, Challenges in the work place, Spirituality.

Unit II: Engineering Ethics, Safety, Responsibilities and Rights.

Overview, Senses of engineering ethics Variety of moral issues, Moral dilemma, Moral autonomy Profession, Models of professional roles, Responsibility, Self-control, Self-interest, Self-respect, Safety definition, Safety and risk, Risk analysis, Confidentiality, Employee rights, Whistle Blowing.

Unit III: Engineering as Social Experimentation

Engineering as experimentation, Engineers as responsible experimenters, Codes of ethics, Industrial standards, A balanced outlook on law, Case-Study.

Unit IV: Global Issues

Globalization, Multinational corporations, Environmental ethics, Computer ethics, Weapons development, Engineers as managers, Engineers as advisors in planning and policy making, Moral leadership.

References:

1. A Textbook on Professional Ethics and Human Values - R. S. Naagarazan, New age international publishers.
2. Human Values and Professional Ethics, Dr. Gurpreet Singh Uppal, 1st edition.
3. Human Values, Tripathi A. N., 3rd edition, New Age International Pvt Ltd Publisher.

Course articulation matrix:

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	1	-	1	2	3	1
CO 2	1	1	1	2	3	1
CO 3	1	1	1	2	3	1
CO 4	1	1	1	2	3	1
Weighted Average	1	1	1	2	3	1

1: Low, 2: Moderate, 3: High

SC PATTERN RECOGNITION [3:1:0]

Objectives:

- Understand pattern recognition systems.
- Learn the different techniques of estimations and component analysis.
- Learn the different supervised & unsupervised learning techniques.

Outcomes:

- Acquire the knowledge on basics of pattern recognition systems
- Demonstrate the techniques of estimations and component analysis.
- Implement different supervised learning techniques.
- Implement different unsupervised learning techniques.

Unit I : Introduction

Machine perception, Pattern recognition systems, Design cycle, Learning and adaptation.

Introduction, Bayesian decision theory - Continuous features, Classifiers Discriminate functions and Decision surfaces, Normal density and Discriminant functions for the Normal Density, Bayes decision theory- Discrete features

Unit II : Maximum Likelihood and Bayesian Parametric Estimation

Introduction, Maximum likelihood estimation, Bayesian estimation, Bayesian parametric estimation, Sufficient statistics, Problems of dimensionality, Component Analysis and Discriminants

Unit III : Nonparametric Techniques

Introduction, Density estimation, Parzen windows, K-Nearest Neighbour estimation, The nearest neighbor rule, Metrics and Nearest Neighbour Classification, Fuzzy Classification, Basics of Neural networks, Support vector machines

Unit IV : Unsupervised Learning

Mixture Densities and Identifiability, Maximum – Likelihood Estimates, Application to Normal Mixtures, Unsupervised Bayesian Learning, Data Description and Clustering, Criterion Functions for Clustering, Hierarchical clustering, Online clustering, Graph Theoretic Methods,

References

1. Pattern Classification, R.O Duda, P.E. Hart and D.G. Stork, 2nd Edition, Wiley publications
2. Pattern Recognition and Image Analysis, Earl Gose, Richard, Johnsonbaugh, Steve Jost, PHI.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	-	-	-	1	-
CO 2	3	-	-	-	3	-
CO 3	3	-	-	-	3	-
CO 4	3	-	-	-	3	-
Weighted Average	3	-	-	-	2.5	-

1: Low, 2: Moderate, 3: High

SC BIG DATA ANALYTICS [3:0:1]

Objectives:

- Understand the Big Data Ecosystem.
- Introduce the students to Hadoop.
- To understand the concepts of Map Reduce and MongoDB
- To understand data Analysis using R

Outcomes:

- Apply the Data Analytics Life Cycle to real life cases.
- Process Data with Hadoop.
- Apply the necessary techniques for data analytics.
- Demonstrate Data Analysis using R.

Unit I: Introduction to Big Data Analytics.

Big Data Overview, State of the Practice in Analytics, Key Roles for the New Big Data Ecosystem, Examples of Big Data Analytics, Data Analytics Lifecycle Overview, Phase 1: Discovery, Phase 2: Data Preparation, Phase 3: Model Planning, Phase 4: Model Building , Phase 5: Communicate Results, Phase 6: Operationalize.

Unit II: Introduction to Hadoop

Introducing Hadoop, Why Hadoop?, Why not RDBMS? RDBMS versus Hadoop, Distributed Computing Challenges, History of Hadoop, Hadoop Overview, Use Case of Hadoop, Hadoop Distributors, HDFS (Hadoop Distributed File System), Processing Data with Hadoop, Managing Resources and Applications with Hadoop YARN (Yet Another Resource Negotiator), Interacting with Hadoop Ecosystem.

Unit III: Introduction to MAPREDUCE Programming and Mongo DB

Introduction, Mapper, Reducer, Combiner, Partitioner, Searching, Sorting, Compression, MongoDB: Uses of MongoDB, Terms Used in RDBMS and MongoDB, Data Types in MongoDB, MongoDB Query Language.

Unit IV: Review of Basic Data Analytic Methods Using R

Introduction to R, Exploratory Data Analysis.

REFERENCES:

1. Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, EMC Education Services, John Wiley & Sons, Inc.
2. Big Data and Analytics, 2ed, Seema Acharya, Subhashini Chellappan, Wiley.
3. Data Science and Analytics, V.K.Jain, Khanna Publishing.
4. Big Data Analytics, M. Vijayalakshmi, Radha Shankarmani, Wiley

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	1	-	1	1
CO 2	-	3	1	-	1	1
CO 3	-	3	1	-	1	1
CO 4	-	3	1	-	1	1
Weighted Average	-	3	1	-	1	1

1: Low, 2: Moderate, 3: High

OE WORLD WIDE WEB [3:1:0]

Objective:

- To provide the conceptual and technological development in the field of Internet and web designing.
- To provide a comprehensive knowledge of Internet, its applications and the TCP/IP protocols widely deployed to provide Internet connectivity worldwide.
- To understand how the World Wide Web with its widespread usefulness has become an integral part of the Internet.
- To provide an overview of basic concepts of web design.

Outcomes:

- Understand the working scheme of the Internet and World Wide Web.
- Evaluate the various protocols of the Internet.
- Comprehend and demonstrate the application of Hypertext Mark-up Language (HTML).
- Apply the various security tools and understand the need of security measures.

Unit I

Internet: introduction, Evolution and History of Internet, Growth of Internet, Internet Services, How Internet Works, Anatomy of Internet, Internet addressing, Internet vs. Intranet, and Impact of Internet.

Unit II

Internet Technology and Protocol: ISO-OSI Reference Model, Data Transmission, Switching, Routers, Gateways, and Network Protocols

Internet Connectivity: Different types of connections, Levels of Internet Connectivity and Internet Service Provider.

Unit III

Web Page Design-HTML: An Introduction, HTML Categories, HTML Fonts, HTML colors, HTML Lists, HTML Tables, HTML Links, HTML Forms, Adding Pictures and Image Attributes.

Unit IV

Computer Networks, Internet & Web Security: Computer Networks, Network Components, Network Topologies, Types of Network Architecture, Network Security, Firewall, Digital Signature, Authentication, Authorization , Copyright issues and Virus.

References

1. Internet Technology and Web Design by Instructional Software Research and Development (ISRD) Group, Tata MC Graw Hill.
2. 2. Programming the World Wide Web, 4th Edition by Robert W. Sebesta.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	**1**	**1**	**1**	**-**	**-**	**-**
CO 2	**1**	**1**	**1**	**-**	**-**	**-**
CO 3	**1**	**1**	**2**	**-**	**1**	**-**
CO 4	**1**	**2**	**2**	**-**	**1**	**-**
Weighted Average	**1**	**1.25**	**1.5**	**0**	**1**	**0**

1: Low, 2: Moderate, 3: High

HC THEORY OF LANGUAGES [3:1:0]

Objectives:

- To learn about the core concepts of automata theory and formal languages.
- To learn fundamentals of Regular and Context Free Grammars and Languages.
- To understand the relation between Regular Language and Finite Automata.
- To understand the relation between Contexts free Languages and PDA.

Outcomes:

- Acquire a fundamental understanding of the core concepts in automata theory and formal languages
- Develop ability to model grammars and automata (recognizers) for different language classes.
- Develop an ability to identify formal language classes and prove language membership properties.
- Develop an ability to prove and disprove theorems establishing key properties of formal languages and automata.

Unit I: Introduction to Automata and Languages

Brief introduction to Formal Proof: Deductive Proofs, Proving equivalences about sets, the contra positive, Proof by contradiction, Counterexamples, Central concepts of automata theory: Alphabets, strings, languages.

Finite Automata: Deterministic Finite Automata, Nondeterministic Finite Automata, Equivalence of DFA and NFA, Finite Automata with Epsilon transitions.

Unit II: Regular Expression and Regular Languages

Regular Expressions, Finite Automata and Regular Expressions: Converting DFAs to regular expressions by eliminating states, converting regular expressions to automata, Applications of regular expressions, Brief overview of algebraic laws of regular expressions.

Properties of Regular Languages : The pumping lemma for regular languages, Applications of the pumping lemma, Closure properties and decision properties of regular languages (proofs not necessary), Minimization of DFAs

Unit III: Context Free Grammars

Context-Free Grammars, Parse Trees, Applications of context-free grammars, Ambiguity in grammars and languages.

Normal Forms of Context-free grammars

Unit IV: Pushdown Automata and Context Free Languages

Pushdown Automata : Definition, Languages of a PDA, Equivalence of PDAs and CFGs, Deterministic Pushdown Automata.

The pumping lemma for context-free languages, Closure properties of context-free languages (proofs not needed).

References

1. “Introduction to Automata Theory, Languages and Computation”, Hopcroft J.E and Ullman,J.D, Narosa Publishing House, Delhi.
2. “Introduction to Languages and Theory of Computation”, John C Martin ,Yd edition, TMH Publication.

3. “Formal Languages and Automata theory”, Basavaraj S. Anami, Karibasappa K G, Wiley India.
4. “Formal Languages and Automata Theory”, C K Nagpal, Oxford University press.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	1	1	-	-	1
CO 2	3	1	1	-	-	1
CO 3	2	1	1	-	-	1
CO 4	3	1	1	-	-	1
Weighted Average	2.75	1	1	0	0	1

1: Low, 2: Moderate, 3: High

HC MACHINE LEARNING [3:0:1]

Objectives:

- Understanding the importance of Machine Learning and demonstrate the use of data frames in Python
- Analyze the process of model building and evaluation
- Comprehend various classification problems
- Discuss the libraries required to implement the techniques of Machine Learning.

Outcomes:

- Identify the need for Machine Learning using Python, appropriate data frames and its operations.
- Ability to build and validate linear regression models
- Ability understand different classification techniques and build classification models
- Ability to use unsupervised learning techniques to cluster data and Apply Scikit library for Machine Learning.

UNIT – I: Introduction to Machine Learning

Introduction to Analytics and Machine Learning, Need for Machine Learning, Framework for Developing Machine Learning Models, Using Python for Machine Learning, Python Stack for Data Science, Getting Started with Anaconda Platform, Introduction to Python.

Descriptive Analytics: Working with Pandas Data Frames in Python, Handling Missing Values, Exploration of Data using Visualization

UNIT – II: Linear Regression

Simple Linear Regression, Steps in Building a Regression Model, Building Simple Linear, Regression Model.

UNIT – III: Classification Problems

Classification Overview, Binary Logistic Regression, Gain Chart and Lift Chart, Classification Tree (Decision Tree Learning).

UNIT – IV: Advanced Machine Learning and Clustering

Scikit-Learn Library for Machine Learning, Advanced Machine Learning Algorithms.

Clustering: Overview, How Does Clustering Work?, K-Means Clustering, Hierarchical Clustering.

References

1. Machine Learning using Python, Manaranjan Pradhan, U Dinesh Kumar, Wiley India Pvt. Ltd., 2019
2. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf, 2013.
3. Learning with Python: How to Think Like a Computer Scientist Paperback – Allen Downey , Jeffrey Elkner, 2015.
4. Jake Vander plas, “Python Data Science Handbook: Essential tools for working with data”, O‘Reilly Publishers,1st Edition.
5. Hands-On Machine Learning with Scikit-Learn and TensorFlow Concepts, Tools, and Techniques to Build Intelligent Systems, Aurelien Geron, O'Reilly Publisher , I edition, 2017

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	1	3	1	-	1	1
CO 2	3	3	1	-	1	1
CO 3	3	3	1	-	1	1
CO 4	3	3	1	-	1	1
Weighted Average	2.5	3	1	-	1	1

1: Low, 2: Moderate, 3: High

HC MINOR PROJECT 0:1:3

Objectives:

- To offer students a glimpse into real world problems and help the students learn how to apply the tools and techniques they learned in the respective courses.
- To help students develop openness to new ideas in computer science and create very precise specifications for the execution of the project idea.
- To promote team working skills, problem solving skills, and presentation skills among students working on the project.

Outcomes:

- Understanding the emerging trends of new technologies by conducting a survey of several available literatures in the preferred field of study.
- Develop real time Projects by comparing the several existing solutions for a research challenge.
- Demonstrate an ability to work in teams and manage the process of building the project within the stipulated time.
- Report and present the findings of the research study/project conducted in the preferred domain.

Students need to implement different kinds of problems using Java based Frameworks, Python, PHP, MYSQL, Cloud tools, IoT tools, Dot NET, CASE tools, Open source tools and Mobile application oriented tools, as well as data mining/machine learning tools and techniques.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	1	3	-	-	-	-
CO 2	1	3	-	-	-	3
CO 3	-	-	3	3	-	3
CO 4	-	-	-	3	2	3
Weighted Average	1	3	3	3	2	3

1: Low, 2: Moderate, 3: High

SC ARTIFICIAL INTELLIGENCE [3:1:0]

Objectives:

- To provide an overview of artificial intelligence (AI) principles and approaches.
- To develop a basic understanding of the building blocks of AI in terms of intelligent agents like Search, Knowledge representation, inference, logic, and learning.
- To provide an overview of knowledge representational structures like slot and fillers.
- To have knowledge of expert systems, learning and planning which plays a considerable role in some applications.

Outcomes:

- Understand the basic concepts of AI.
- Understand the fundamentals of knowledge representation, inference and theorem proving.
- Represent knowledge of the world using logic and infer new facts from that knowledge.
- Explain how Artificial Intelligence enables capabilities that are beyond conventional technology.

Unit I: Introduction

AI Problems, AI Techniques, Defining the Problem as State Space Search, Production Systems, Problem Characteristics, Production System Characteristics, Issues in the Design of Search Programs.

Unit II: Heuristic Search Techniques and Knowledge Representation

Generate and Test, Hill climbing, BFS, DFS, Knowledge Representation Issues, Approaches to Knowledge Representation, Procedural Versus Declarative Knowledge, Inferential Versus Inheritable Knowledge, Normal Forms in Predicate Logic and Clausal Forms, Introduction to Non-monotonic Reasoning, Logics for Non-monotonic Reasoning.

Unit III: Knowledge Representational Structures

Weak Slot and Filler Structures: Semantic Nets, Frames.

Strong Slot and Filler Structure: Conceptual Dependency, Scripts.

Unit IV: Game Playing, Planning and Expert Systems

Game Playing: Minimax Search Procedure, Adding Alpha-Beta Cut Offs, Planning-Goal Stack Planning, Expert Systems: Expert System Versus Conventional Computer, Expert System Shells, and Explanation Based Learning.

References

1. “Artificial Intelligence”, Rich Elaine Knight Kevin – Tata McGraw Hill.
2. “Introduction to Artificial Intelligence and Expert system”, Patterson W Dan – Prentice Hall.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	3	2	2	2	3
CO 2	3	3	3	2	1	3
CO 3	3	3	3	2	1	3
CO 4	3	3	3	2	1	3
Weighted Average	3	3	2.75	2	1.25	3

1: Low, 2: Moderate, 3: High

SC DIGITAL IMAGE PROCESSING [3:0:1]

Objectives:

- Understand the fundamentals of digital image processing.
- Learn the different Image enhancement techniques.
- Understand the image segmentation techniques.

Outcomes:

- Demonstrate the concepts of digital image processing.
- Learn different techniques employed for the enhancement of images using spatial domain.
- Learn different techniques employed for the enhancement of images using frequency domain.
- Implement the techniques of image segmentation.

Unit I: Introduction and Digital Image Fundamentals

Introduction to Digital Image Processing, The Origins of Digital Image Processing, Examples of Fields that use Digital Image Processing, Fundamental steps in Digital Image Processing, Components of Image Processing System, Elements of Visual Perception, Image Sampling and Quantization, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations

Unit II: Image Enhancement in the Spatial Domain

Some Basic Gray Level Transformations, Histogram Processing, Enhancement using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters.

Unit III: Image Enhancement in the Frequency Domain

Introduction to the Fourier Transform and the Frequency Domain, Smoothing Frequency Domain Filters, Sharpening Frequency Domain Filters, Homomorphic Filtering.

Unit IV: Image Segmentation

Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-based Segmentation, Segmentation by Morphological Watersheds.

Reference

1. Digital Image Processing – Rafel C. Gonzalez and Richard E. Woods, 2nd Edition, Pearson Education.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	-	-	-	2	-
CO 2	3	-	-	-	3	-
CO 3	3	-	-	-	3	-
CO 4	3	-	-	-	3	-
Weighted Average	3	-	-	-	3.5	-

1: Low, 2: Moderate, 3: High

SC C# PROGRAMMING [3:0:1]

Objectives:

- To provide an overview of the .NET framework.
- Understand Object-Oriented Paradigm using C# programming.
- Learn extended OOP's concept in C# environment.
- Understand the concepts of interfaces and multithreading.

Outcomes:

- Acquire the knowledge of .NET framework.
- Develop an ability to write programs in C#.
- Implement the extended OOP's concept in C# environment.
- Develop applications using standard C# libraries.

Unit I

Understanding .NET: The C# Environment: The .Net Strategy, The Origins of .Net Technology, The .NET Framework, The Common Language Runtime, Framework Base Classes, Benefits of the .NET Approach.

Overview of C#: Introduction, A Simple C# Program, Namespaces, Adding Comments, main Returning a Value, Using Aliases for Namespace Classes, passing String Objects to Write Line Method, Command Line Arguments, Main with a Class, Providing Interactive Input, Using mathematical Functions, Multiple main Methods, Compile Time Errors, Program Structure, Program Coding Style.

Methods in C#

Introduction, Declaring Methods, The Main Method, Invoking Methods, Nesting of Methods, Method Parameters, Pass by Value, Pass by Reference, The Output Parameters, Variable Argument Lists, Method Overloading.

Arrays, Strings, Structures and Enumerations.

Unit II

Classes and Objects: Introduction, Basic Principles of OOP, Defining a Class, Adding Variables, Adding Methods, Member Access Modifiers, Creating Objects, Accessing Class members, Constructors, Static Members, Static Constructors, Private Constructors, Copy Constructors, Destructors, Member Initialization, The this Reference, Nesting of Classes, Constant Members, Read-only Members, Properties, Indexers.

Operator Overloading: Introduction, Over loadable Operators, Need for Operator Overloading, Defining Operator Overloading, Overloading Unary Operators, Overloading Binary Operators, Overloading Comparison Operators.

Unit III

Inheritance: Introduction, Classical Inheritance, Containment Inheritance, Defining a Subclass, Visibility Control, Defining Subclass Constructors, Multilevel Inheritance, Hierarchical Inheritance.

Run-Time Polymorphism: Overriding methods, Hiding Methods, Abstract Classes, Abstract Methods, Sealed Classes, and Sealed Methods.

Managing Errors and Exceptions: Introduction, Debugging, Types of Errors, Exceptions, Syntax of Exception handling Code, Multiple Catch Statements, Using Finally Statements, Nested Try Blocks, Throwing Our Own Exceptions, Checked and Unchecked Operators.

Unit IV

Interfaces: Introduction, Defining an Interface, Extending an Interface, Implementing Interfaces, Interfaces and Inheritance, Abstract Class and Interfaces.

Multithreading in C#: Introduction, Understanding the System. Threading Namespace, Creating and Starting a Thread, Scheduling a Thread, Synchronizing Threads, Thread Pooling.

Delegates and Events: Introduction, Delegates, Delegate Declaration, Delegate Methods, Delegate Instantiation, Delegate Invocation, Multicast Delegates, Events.

References

1. PROGRAMMING IN C# - A PRIMER by E Balaguruswamy, Third Edition, and Tata McGraw-Hill.
2. C# 4.0: The Complete Reference by Herbert Schildt, Tata McGraw-Hill.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	1	-	-	2
CO 2	-	2	1	-	-	1
CO 3	-	3	2	-	-	1
CO 4	-	3	2	-	-	2
Weighted Average	-	2.75	1.5	-	-	1.5

1: Low, 2: Moderate, 3: High

SC ANDROID PROGRAMMING [3:0:1]

Objectives:

- Learn to build simple android applications.
- Get an understanding of essentials of application design and user interface design.
- Understand different android APIs used to store and manage the data through SQLite.
- Understanding different android networking and web APIs to share the data between the applications.

Outcomes:

- Build sample android application.
- Develop user interfaces for android applications.
- Develop android applications to share data between different applications.
- Deploy android applications.

Unit I: Introduction to Android

History of Mobile Software Development, The Open Handset Alliance, The Android Platform Android SDK, Building a sample Android application, Anatomy of Android applications, Android terminologies.

Unit II: Android Application Design Essentials

Application Context, Activities, Services, Intents, Receiving and Broadcasting Intents, Android Manifest File and its common settings , Using Intent Filter, Permissions , Managing Application resources in a hierarchy , Working with different types of resources.

Unit III: Android User Interface Design Essentials

User Interface Screen elements, Designing User Interfaces with Layouts, Drawing and Working with Animation.

Unit IV: Using Android APIs

Using Android Data and Storage APIs, Managing data using SQLite, Sharing Data between Applications with Content Providers , Using Android Networking APIs , Using Android Web APIs , Using Android Telephony APIs , Deploying (selling) your Android application

References:

1. “Android Wireless Application Development”, Lauren Darcey and Shane Conder, 2nd edition, Pearson Education.
2. “Professional Android 2 Application Development”, Reto Meier, Wiley India.
3. “Beginning Android”, Mark L Murphy, Wiley India.
4. “Pro Android”, Sayed Y Hashimi and Satya Komatineni, Wiley India.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	2	3	2	-	1	1
CO 2	3	3	2	-	1	2
CO 3	3	3	2	-	2	2
CO 4	3	3	3	-	2	2
Weighted Average	2.75	3	2.25	-	1.5	1.75

1: Low, 2: Moderate, 3: High

SC SOFTWARE ENGINEERING [3:1:0]

Objectives:

- Understand the importance of domain knowledge and its work around.
- Know the importance team work and stewardship.
- Understand the phases in a software project.
- Understand the fundamental concepts of requirements engineering and Analysis Modelling.
- Learn various testing and maintenance measures.

Outcomes:

- Identify the key activities in managing software project and compare different process models.
- Able to develop software using contemporary agile approaches
- Gain the ability to work as an individual and as part of a multidisciplinary team to develop and deliver quality software.
- Compare and contrast the various testing and maintenance approaches.

Unit I: Software, Software Engineering and Process Models

The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths, Prescriptive Process Models.

Unit II: Agile Development

Introduction to Agile development, Agility and Cost of Change, Agile Process, Extreme Programming, User stories, Brief introduction to Scrum.

Unit III: Requirements Modeling & Design

Requirements Analysis, Scenario – Based Modeling, UML Models that supplement the Use Case, Data Modeling Concepts, Requirements Modeling Strategies, Flow-oriented Modeling, Creating a behavioural model, Design concepts, Design Model.

Unit IV: TESTING

Software Quality Assurance: Elements of Software Quality Assurance, Software testing fundamentals-Internal and external views of Testing-white box testing- basis path testing-control structure testing-black box testing- Regression Testing – Unit Testing – Integration Testing – Validation Testing – System Testing and Debugging.

References

1. “Software Engineering – A Practitioner’s Approach”, Roger S. Pressman, Seventh Edition, Mc Graw-Hill.
2. “Software Engineering”, Ian Sommerville, 9th Edition, Pearson Education Asia.
3. “Fundamentals of Software Engineering”, Rajib Mall, Third Edition, PHI.
4. “Software Engineering - A Precise Approach”, Pankaj Jalote, Wiley India.
5. “Software Engineering”, Kelkar S.A.,PHI.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	3	1	1	-	1
CO 2	-	3	2	2	-	1
CO 3	-	2	2	2	-	-
CO 4	-	1	2	-	-	1
Weighted Average	0	2.25	1.75	1.25	0	1

1: Low, 2: Moderate, 3: High

OE E-COMMERCE [3:1:0]

Objectives:

- To impart knowledge on E-Commerce.
- To provide an overview of various applications connected with E-Commerce.
- To enable the learner for aiming careers in special software development involving E-Commerce technologies.
- Understand the security issues in E – commerce.

Outcomes:

- Study the impact of E-commerce on business models and strategy
- Describe the Internet trading relationships including Business to Consumer, Business-to-Business, Intra-organizational structures.
- Assess electronic payment systems and its securities.
- Recognize, discuss and derive possible solutions to global E-commerce issues.

Unit 1: Introduction to E-Commerce

Definition, Scope of E-Commerce, Hardware requirements, E-Commerce and Trade Cycle, Electronic Markets, Electronic Data Interchange and Internet Commerce.

Unit 2: Business to Business E-Commerce

Electronic Markets, Electronic Data Interchange (EDI): Technology, Standards (UN/EDIFACT), Communications, Implementations, Agreements, Security, EDI and Business, Inter-Organizational Ecommerce. Business models for E-commerce, Business Process Re-Engineering.

Unit 3: Business to Consumer E-Commerce and E-Business

Consumer trade transaction, Web metrics, Elements of E-Commerce, Industry impacts of E-business. Integrating Intranet and internet web applications across multiple networks. Internet bookshops, Software supplies and support, Electronic Newspapers, Internet Banking, Virtual Auctions, Online Share Dealing, Gambling on the net, E-Diversity, Case studies through internet.

Unit 4: Security Issues

How criminals plan attacks, passive attack, Active attacks, cyber stalking, Secure Electronic Transaction (SET) Protocol, Electronic cash over internet, Internet Security, Search engines, Intelligent agents in E-Commerce Electronic payment systems

References

1. E-Commerce: Strategy, Technologies & Applications, David Whitley, McGraw Hill.
2. E-commerce: The Cutting Edge of Business, K. K. Bajaj and Debjani Nag, 2nd Edition, McGraw Hill.
3. Handbook of Electronic Commerce, Shaw et al., Springer.
4. Global Electronic Commerce- Theory and Case Studies, C. Westland and T. H. K. Clark, University Press.
5. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Sunit Belapure and Nina Godbole, Wiley India.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	1	2	2	2	-
CO 2	-	2	3	2	2	-
CO 3	-	2	2	1	1	-
CO 4	-	-	2	2	1	-
Weighted Average	0	1.25	2.25	1.75	1.5	0

1: Low, 2: Moderate, 3: High

HC DISSERTATION [0:2:10]

Objectives:

- Able to design a computing system to meet desired needs within realistic constraints such as safety, security and applicability.
- An ability to conduct experiments, interpret data and provide well informed conclusions.
- An ability to select modern computing tools and techniques and use them with dexterity.

Outcomes:

- Develop basic algorithm steps as a solution to a real-life problem.
- Implement algorithms using latest tools that contribute to the software solution of the project using different tools.
- Analyse, interpret, test and validate experimental results.
- Develop research/technical report with enhanced writing/communication skills following ethical practices.

Students need to implement different kinds of problems using Java based Frameworks, Python, PHP, MYSQL, Cloud tools, IoT tools, Dot NET, CASE tools, Open source tools and Mobile application oriented tools, as well as data mining/machine learning tools and techniques.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	3	2	-	-	-
CO 2	-	-	3	-	3	-
CO 3	1	3	-	-	2	-
CO 4	-	-	3	3	3	3
Weighted Average	2	3	2.66	3	2.66	3

1: Low, 2: Moderate, 3: High

SC COMPILER CONSTRUCTION [3:1:0]

Objectives:

- To introduce principal structure of compiler, basic theories and methods used for different parts of compiler.
- To impart knowledge of fundamentals of language translator, structure of a typical compiler, parsing methods etc.
- To design various phases of compiler such as Lexical analyser, parser etc.
- To distinguish different optimization techniques in the design of compiler.

Outcomes:

- Explain the concepts and different phases of compilation and Interpret language tokens using regular expressions and design lexical analyzer.
- Build top down parsing, bottom up parsing and parse tree representation of the input.
- Perform context sensitive analysis, semantic analysis and type checking
- Experiment the optimization techniques to intermediate code and generate machine code for high level language program.

Unit 1

Introduction to compiler, A high level view of compilation, General Structure of a compiler, an overview of compilation technology.

Introduction and Lexical Analysis (Scanning)

Regular Languages/Expressions, finite state machines, building regular expressions from finite automation.

Unit 2

Syntax Analysis (Parsing)

Expression Syntax, Context Free Grammers, Top-Down Parsing, Bottom-Up Parsing.

Unit 3

Semantic Analysis

Context-Sensitive Analysis, Attribute Grammers, Symbol Tables, Type Checking.

Unit 4

Intermediate Representations

Properties, taxonomy, graphical IRs, Linear IRs, storage management, the procedure abstraction, linkage convention, run-time storage organization, code optimization, code generation

Reference:

1. Compilers, principles, techniques and tools, Aho, A.V., Sethi R and Ullman J.,D., Addison Wesley.
2. “Engineering a compiler”, Keith Cooper, Linda Torczon, Morgan Kaufmann.
3. The Essence of Compilers, Hunter R., Prentice Hall.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	2	3	-	-	-	1
CO 2	3	3	-	-	-	1
CO 3	3	3	-	-	-	1
CO 4	2	3	-	-	-	1
Weighted Average	2.5	3	0	0	0	1

1: Low, 2: Moderate, 3: High

SC ADVANCED DATABASE MANAGEMENT SYSTEM [2:1:1]

Objectives:

- To evaluate emerging architectures for database management systems.
- To develop an understanding the manner in which relational systems are implemented and the implications of the techniques of implementation for database performance.
- To assess the impact of emerging database standards on the facilities which future database management systems will provide.

Outcomes:

- Critically assess new developments in database technology.
- Evaluate the contribution of database theory to practical implementations of database management systems.
- Implement the various types of database systems.
- Interpret the impact of emerging database standards.

Unit I: Database Design Methodology, Query Processing and Physical Design

Database Design and Implementation process, UML diagrams as an aid to Database Design Specification, Overview of Query Processing : Measures of Query cost, Algorithms for SELECT and JOIN Operations, Pipelining : Implementation of Pipelining, Evaluation algorithms for pipelining, Overview of Query Optimization, Physical Database Design in Relational Databases.

Unit II: Transaction Processing Concepts, Object and Object-Relational Databases

Introduction to Transaction Processing: Transaction and System Concepts, Desirable Properties of Transactions, Transaction Support in SQL.

Concepts for Object Databases: Overview of Object-Oriented Concepts, Object Identity, Object Structure, and Type Constructors, Encapsulation of Operations, Methods, and Persistence, Type Hierarchies and Inheritance. Overview of the Object Model of ODMG, Overview of SQL and its Object-Relational Features, Evolution of Data Models and Current Trends.

Unit III: Security, Advanced Modelling and Distribution

Database Security : Security issues, Enhanced Data Models for Advanced Applications: Active Database Concepts and triggers, Distributed Databases: Distributed Database Concepts, Data Fragmentation, Transparency, Distributed Transactions, Types of Distributed Database Systems, Overview of Concurrency Control Distributed Databases.

Unit IV: Emerging Technologies

Overview of Data Mining Technology, Emerging Database Technologies and Applications: Mobile Databases, Multimedia Databases, Geographic Information Systems (GIS).

References:

1. Fundamentals of Database Systems – Fifth Edition – Ramez Elmasri, Shamkant B Navathe.
2. Database System Concepts – Abraham Siberschatz, Henry F. Korth, S. Sudarshan, Fifth Edition- McGraw – Hill.
3. Database Systems – Thomas Connolly, Carolyn Becg – Third Edition – Pearson Education.
4. An Introduction to Database Systems – Eight Edition- Date C J - Addison Wesley.
5. Strategic Database Technology – Simon A R, Morgan Kaufmann.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	2	1	-	1	1
CO 2	3	3	2	-	2	2
CO 3	3	3	3	-	2	3
CO 4	3	3	2	-	2	2
Weighted Average	3	2.75	2	0	1.75	2

1: Low, 2: Moderate, 3: High

SC DATA MINING [3:0:1]

Objectives:

- To get an understanding of methods and applications of Data mining.
- Understand the rules related to association, classification and clustering analysis.
- Compare and contrast between different classification and clustering algorithms

Outcomes:

- Identify data mining problems and recognise types of data and preprocessing needed.
- Employ the concepts of Association Analysis
- Identify problems suitable for Classifications and Apply different classification algorithms
- Identify problems appropriate for Clustering and Apply different clustering algorithms.

Unit I:

Introduction to data mining, Challenges, Data Mining Tasks.

Data: Types of Data, Data Quality, Data Pre-processing, Measures of Similarity and Dissimilarity.

Unit II:

Association Analysis: Association Analysis: Problem Definition, Frequent Item set Generation, Rule generation. Alternative Methods for Generating Frequent Item sets, FP-Growth Algorithm, Evaluation of Association Patterns.

Unit III:

Classification: Decision Trees Induction, Method for Comparing Classifiers, Rule Based Classifiers, Nearest Neighbor Classifiers, Bayesian Classifiers.

Unit IV:

Clustering Analysis: Overview, K-Means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster Evaluation, Density-Based Clustering, Graph Based Clustering, Scalable Clustering Algorithms.

References:

1. “Introduction to Data Mining”, Pang-Ning Tan, Michael Steinbach, Vipin Kumar Pearson.
2. “Data Mining -Concepts and Techniques”, Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann Publisher, 3rd Edition.
3. “Mastering Data Mining” Michael.J.Berry, Gordon.S.Linoff, Wiley Edition, second edition.
4. “Principles of Data Mining”, David Hand, Heikki Mannila and Padhraic Smyth, The MIT Press.
5. “Data Mining Techniques”, Arun K Pujari, University Press.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	3	1	-	2	1
CO 2	3	3	1	-	2	1
CO 3	3	3	1	-	2	1
CO 4	3	3	1	-	2	1
Weighted Average	3	3	1	0	2	1

1: Low, 2: Moderate, 3: High

OE OFFICE AUTOMATION [3:1:0]

Objectives:

- To provide a basic introduction to computers and computing environment.
- To enable the students in crafting professional documents using word pre-processors.
- To enable students use spreadsheets for tabulating and calculating data and create graphical representations of data.
- To enable students to design professional presentations.

Outcomes:

- To understand the basics of computer hardware and software.
- To prepare documents of different types.
- Ability to develop and use spreadsheets for tabulating and analysing for productivity.
- To prepare presentations.

Unit I

Introduction to Computers, Basic Anatomy of Computers and Introduction to MS-Office.

Unit II

MS-Word – Word Basics, Formatting Features, Menu, Commands, Tool Bars and their Icons, Mail Merge and Macros Creating Tables.

Unit III

MS-Excel - Introduction, Menu, Commands, Tool Bars and their Icons, and Functions.

Unit IV

MS-Power Point – Menu, Toolbar, Navigating in PowerPoint, Working with PowerPoint and Introduction to MS-Access.

References:

1. MS Office for Everyone – Sanjay Sanena, Vikas Publishing House.
2. Step by Step Microsoft Office XP, PHI.

Course Articulation Matrix

PO CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	2	2	1	1	1	1
CO 2	2	2	1	1	1	1
CO 3	2	2	1	1	1	1
CO 4	2	2	1	1	1	1
Weighted Average	2	2	1	1	1	1

1: Low, 2: Moderate, 3: High

SBRR Mahajana First Grade College (Autonomous), PG Wing

Accredited by NAAC with A Grade

Pooja Bhagavat Memorial Mahajana Education Centre,

Affiliated to University Of Mysore

M.Sc. in Computer Science

QUESTION PAPER PATTERN

Duration: **3 Hours** Max Marks: **70**

(There are 5 questions. All questions must be answered.)

Question 1

There are 10 objective questions each carries 1 mark. 10 ×1=10

Question 2

There are 2 main questions (a) and (b) each carrying 15 marks. Candidate has to answer any one (a or b). This covers unit 1 of the syllabus. (Each main question can be split into sub- questions totalling 15 marks) 15 ×1=15

Question 3

There are 2 main questions (a) and (b) each carrying 15 marks. Candidate has to answer any one (a or b). This covers unit 2 of the syllabus. (Each main question can be split into sub- questions totalling 15 marks) 15 ×1=15

Question 4

There are 2 main questions (a) and (b) each carrying 15 marks. Candidate has to answer any one (a or b). This covers unit 3 of the syllabus. (Each main question can be split into sub- questions totalling 15 marks) 15 ×1=15

Question 5

There are 2 main questions (a) and (b) each carrying 15 marks. Candidate has to answer any one (a or b). This covers unit 4 of the syllabus. (Each main question can be split into sub- questions totalling 15 marks) 15 ×1=15

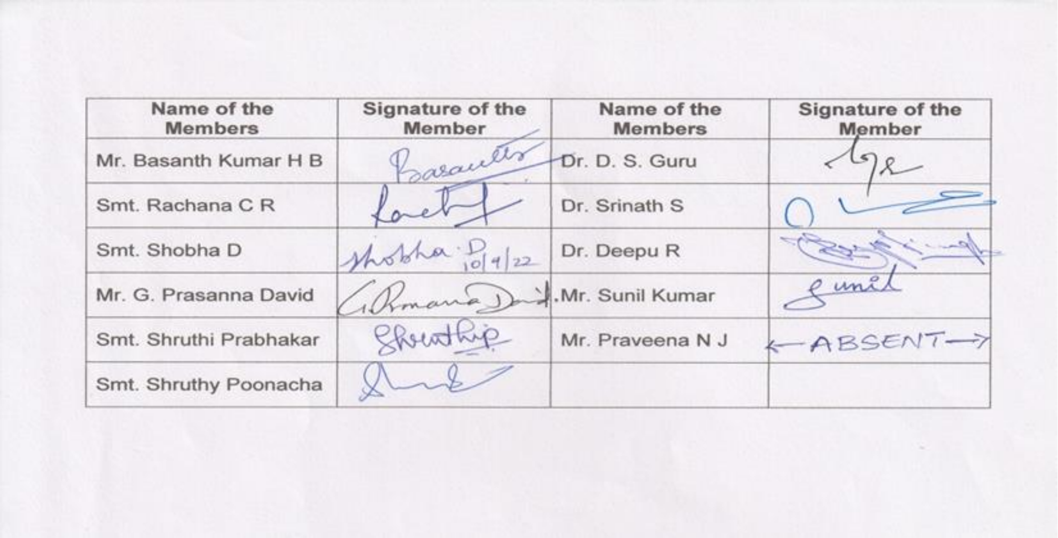
****

Department: M.Sc. Computer Science
Board of Studies

Sl. No.	Category	Name	Designation	Address for communication	e-Mail and Mobile No.
1	Chairperson	Mr. Basanth Kumar H B	Assistant Professor	Department of Studies in Computer Science, SBRR Mahajana First Grade College, PG Wing, KRS Road, Metagalli, Mysuru – 570016.	basanth.10@gmail.com 9611882250
2	Faculty of the department	Smt. Rachana C R	Associate Professor & Head		rachanacr@gmail.com 8095645644
		Smt. Shobha D	Assistant Professor		shobhadgps@gmail.com 8867769209
		Mr. Prasanna David	Assistant Professor		prasannadavid@gmail.com 9980095772
		Smt. Shruthi Prabhakar	Assistant Professor		shruthiprabhakar412@gmail.com 9980715577
		Smt. Shruthy Poonacha S	Assistant Professor	Computer Science Department, S.B.R.R. Mahajana First Grade College, Jayalakshmipuram, Mysuru – 570012.	shruthypoonacha@gmail.com 9886367273
3	Two Experts from Outside the College	Dr. Deepu R	Professor	Department of CSE, ATME College of Engineering, Mysuru-Kanakapura-Bangalore Road, Mysuru– 570028.	dr.deepur_cs@atme.edu.in 9845490505
		Dr. Srinath S	Associate Professor & Head of the Department	Department of CSE, SJCE, JSS Science and Technology University,	srinath@sjce.ac.in 9844823201

				Mysuru-570006.	
4	Nominee by the Vice Chancellor	Dr. D. S. Guru	Professor	Department of Studies in Computer Science, University of Mysore, Mysuru.	dsg@compsci.uni-mysore.ac.in 9620228005
5	One Person from Industry / Corporate Sector / Allied area	Mr. Sunil Kumar	Managing Director	#103 MIG, Kamakshi Hospital Road, Block I, Ramakrishnanagar, Mysuru, Karnataka 570023.	suniln@intrella.in 9036453696
6	Alumnus	Mr. Praveena N J	Senior Engineer	#784, 6th Cross, Paduka Mandir Road, Uttarahalli, Bengaluru.	praveennj4@gmail.com 9916096159

APPROVED BY THE FOLLOWING BoS MEMBERS



Name of the Members	Signature of the Member	Name of the Members	Signature of the Member
Mr. Basanth Kumar H B	Basanth	Dr. D. S. Guru	
Smt. Rachana C R		Dr. Srinath S	
Smt. Shobha D	Shobha · D 10/4/22	Dr. Deepu R	
Mr. G. Prasanna David	G Prasanna David	Mr. Sunil Kumar	Sunil
Smt. Shruthi Prabhakar	Shruthi P	Mr. Praveena N J	← ABSENT →
Smt. Shruthy Poonacha			