SCHEME AND SYLLABUS OF M Sc COMPUTER SCIENCE
(2016 ADMISSION ONWARDS)

A. Objectives

1. To develop interest among the candidates towards a career in academics and research, and to enable them with sufficient knowledge to become a competent academician.
2. To equip the students with sufficient exposure and skills to enable them to catch a deserving position in software industry.
3. To develop interest in promoting the use of Computer Science for the positive development of our society and the environment.
4. To enable the students to contest for national level competitive examinations.

B. Eligibility

Candidates for admission to M Sc Programme in Computer Science should have passed

(i) A Degree course with minimum 3 years duration after 10+2 with not less than 50% marks or 2 CGPA[S] out of 4 in Computer Science/Computer Application/Electronics as main or an equivalent Degree recognized by the University of Kerala for the purpose.

(ii) Any science degree with minimum 3 years duration after 10+2 with not less than 50% marks or 2 CGPA[S] out of 4 with Computer science/Computer application as one of the main/subsidiary/core subject.

Candidate shall meet all other requirements in the prospectus published by University time to time.

C. Evaluation

Evaluation will be done in two components: Continuous Evaluation (CE) and End-Semester Evaluation (ESE).

(i) Continuous Evaluation (CE)

Theory Courses: In addition to class room lectures, students shall be assigned with application problems, class room presentations, group activities etc. Case studies/industry visits may also be organized. At least two tests shall be conducted for each course. Short viva may be conducted to assess assignments.

CE Mark for Theory (out of 25)

| Assignments and Activities | 10 (distributed for minimum two components) |
| Test | 10 Attendance : 5 |

Lab Courses: Each Lab course shall be completed under the supervision of a faculty member. The students shall undertake a case study for each practical course. The case study can be done as a team of 2 members if necessary. The practical record includes both lab exercises and case study report.

CE Mark for Lab Courses (out of 25)

| Attendance | 5 Lab performance : 5 Record : 5 |
| Internal Lab Test | 5 Case Study : 5 |

Minor Project & Seminar : Minor project shall be done in the college itself, under the guidance of a faculty in the department. The project can be done individually, or as a team of two members. The volume of work shall be limited to be completed in not more than 50 hrs.

CE Mark for Minor Project & Seminar (out of 50)

CE marks for Minor project & seminar put together out of 50 shall be submitted to the University. However, the faculty in charge shall assign the marks as follows:
(a) Minor Project:

<table>
<thead>
<tr>
<th>Activity</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report</td>
<td>5</td>
</tr>
<tr>
<td>Implementation/Findings</td>
<td>5</td>
</tr>
<tr>
<td>Design and development</td>
<td>10</td>
</tr>
<tr>
<td>Presentation and Defense</td>
<td>5</td>
</tr>
</tbody>
</table>

(b) Seminar:

<table>
<thead>
<tr>
<th>Activity</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Presentation &amp; Defense</td>
<td>12</td>
</tr>
<tr>
<td>Report</td>
<td>8</td>
</tr>
<tr>
<td>Topic &amp; content organization of presentation</td>
<td>5</td>
</tr>
</tbody>
</table>

Major Project:

Major project work shall be done individually by each student under the guidance of a faculty member from the department. An internal evaluation team consisting of at least three members, chaired by the Head of Department or a senior faculty member shall be constituted at the college every year by the end of third semester. The project guide of the candidate can be one of the members in the team.

If the student chooses to do his project in an organization other than the college, the department shall ensure the following:

(i) The project is supervised by a qualified person. The External Supervisor shall be a post graduate in either Science/Applied Science/Engineering branches. He/She shall have at least 3 years experience in running/managing/implementing/supervising such projects. A declaration shall be obtained in this regard from that person, and shall be kept with the Department.

(ii) An attendance statement and a performance feedback shall be obtained from the External supervisor. The student has to present periodic reports and attend for evaluation process before the internal evaluation team at the College as per the schedule.

CE Marks for Major Project (out of 100)

<table>
<thead>
<tr>
<th>Activity</th>
<th>Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Study Phase activities &amp; Report/Literature survey</td>
<td>20</td>
</tr>
<tr>
<td>Design</td>
<td>20</td>
</tr>
<tr>
<td>Methodology</td>
<td>20</td>
</tr>
<tr>
<td>Findings/Implementation</td>
<td>20</td>
</tr>
<tr>
<td>Presentation &amp; Defense</td>
<td>20</td>
</tr>
</tbody>
</table>

(ii) End Semester Evaluation (ESE)

End-semester evaluation for all courses will be conducted by the University. A student with 75% attendance in a course is eligible to attend the University examination.

Theory Courses: The question paper consists of two parts:

Part A (27 marks). 9 compulsory questions, of 3 marks each.

Part B (48 marks). Students must answer two out of three questions from each module. Each question carries 8 marks.

Lab Courses: Lab examinations shall be conducted in each college by two examiners appointed by the University, of which one shall be from other colleges. The External Examiner will finalize the marks in consultation with the internal examiner. The questions for the examination shall be prepared before each examination and approved by the board of examiners. A candidate shall be asked to answer one out of two questions given to him.
The marks will be distributed as follows. (Total 75)

Description of procedure: 10 marks [The procedure/algorithm/flow chart/pseudo code for solving the problem(s) shall be explained in the answersheet.]

Preparation of program: 15 marks

Logic & Output : 20 marks. 20 marks shall be distributed as follows:

(i) 15 marks for the correct output of the given problem.

(ii) 5 marks for completing the modifications suggested by the examiner(s) in the given questions during the examination hours.

[Viva: 15 marks. Viva shall be on the problem domain, based on the programming tool used, from the area of study for solving the problem/from the theory concepts related to the area.]

Case study Report and Viva : 15 marks

Minor Project Total : 50

Minor Project shall be evaluated by an examiner appointed by the university. A report of the project shall be submitted to the examiner at the time of examination.

<table>
<thead>
<tr>
<th>Report of the work</th>
<th>10 marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methodology and Topic</td>
<td>5 marks</td>
</tr>
<tr>
<td>Findings/Implementation</td>
<td>10 marks</td>
</tr>
<tr>
<td>Presentation &amp; Defense</td>
<td>15 marks</td>
</tr>
<tr>
<td>Viva Voce</td>
<td>10 marks</td>
</tr>
</tbody>
</table>

Major Project Total : 100

Major Project shall be evaluated at the examination centres by a panel of two examiners appointed by the university, one of which shall be from other colleges. The project report shall be finalized after the internal evaluation. The candidates shall present the findings/output of their work before the examiners during the examination hours. The examiners will conduct a viva voce also.

<table>
<thead>
<tr>
<th>Report of the work</th>
<th>20 marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methodology and Content</td>
<td>25 marks</td>
</tr>
<tr>
<td>Findings/Implementation</td>
<td>20 marks</td>
</tr>
<tr>
<td>Presentation &amp; Defense</td>
<td>20 marks</td>
</tr>
<tr>
<td>Viva Voce</td>
<td>15 marks</td>
</tr>
</tbody>
</table>

Comprehensive Viva

It is mandatory that the Comprehensive Viva shall be conducted by separate examiners than that for Project Evaluation. The viva will be conducted by a panel of two examiners appointed by the University, of which one shall be from outside the college. Though the viva shall be based on the entire syllabus contents, the candidates may be given opportunity to opt a set of subjects, not less than 40% of the programme. However, the candidate, in any case, shall not be asked to write answers to questions given by the examiners.

D. Pass Requirements:

For each subject(including practical), a student should get a minimum of 40% marks for the university examinations and 50% aggregate for the CE and ESE together for all theory and practical courses except Major Project. For Major Project and Comprehensive viva-voce in the 4th semester each student should get a minimum of 50% for the university examination and 50% aggregate for the CE and ESE together. Classification of passed candidates will be as per the University norms.
### E. SCHEME

<table>
<thead>
<tr>
<th>Semester I</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CE</th>
<th>ESE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1611 Computer Architecture</td>
<td>3</td>
<td>-</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1612 Data Structures and Algorithms</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1613 Mathematical Foundations of Computer Science</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1614 Programming Paradigms</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1615 Computer Networks</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1616 Data Structures &amp; Algorithms Lab</td>
<td>3</td>
<td>25</td>
<td></td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>CS1617 Java Programming Lab</td>
<td>3</td>
<td>25</td>
<td></td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester II</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CE</th>
<th>ESE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1621 Modern Operating Systems</td>
<td>3</td>
<td>-</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1622 Advances in Database Management</td>
<td>3</td>
<td>-</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1623 Object Oriented Analysis and Design</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1624 Graphics &amp; Multimedia Systems</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1625 Optimization Techniques</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1626 Minor Project &amp; Seminar</td>
<td>4</td>
<td></td>
<td></td>
<td>50</td>
<td>50</td>
<td>100</td>
</tr>
<tr>
<td>CS1627 Database &amp; Web Programming Lab</td>
<td>3</td>
<td>25</td>
<td></td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester III</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CE</th>
<th>ESE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1631 Data Mining &amp; Warehousing</td>
<td>3</td>
<td>-</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1632 Distributed Systems and Cloud Computing</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1633 Information Security</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1634 Compiler Design</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1635 Elective I</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1636 Network Administration Lab</td>
<td>3</td>
<td>25</td>
<td></td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>CS1637 Distributed Computing Lab</td>
<td>3</td>
<td>25</td>
<td></td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Semester IV</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>CE</th>
<th>ESE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1641 Research &amp; Technical Writing</td>
<td>3</td>
<td>1</td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1642 Elective II</td>
<td>3</td>
<td></td>
<td></td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>CS1643 Major Project</td>
<td>18</td>
<td>100</td>
<td></td>
<td>100</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>CS1644 Comprehensive Viva Voce</td>
<td>—</td>
<td></td>
<td></td>
<td>—</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

**Total** 2600

**Elective I:**
- A. Big Data Analytics
- B. Digital Image Processing
- C. Machine Intelligence
- D. Programming for Portable Devices

**Elective II**
- A. Parallel Computing Algorithms
- B. Bioinformatics
- C. Neural Networks & Fuzzy Systems
- D. Embedded Systems
SEMESTER I
CS1611 Computer Architecture

Module I: Basic Structure of digital Computer, review of digital fundamentals, combinatorial circuits (review only). Central Processing Unit: General Register and Stack Organization, instruction set architecture, examples of assembly language instructions (8085), Addressing Modes, Program Control, CISC, RISC, pre-fetching of microinstructions, Parallel Processing Architectures - Parallelism in sequential Machines, Multiprocessor Architecture. Pipelining and Super Scalar Techniques - Linear Pipeline Processors.

Module II: Memory Organization, Memory system considerations, cache memory, virtual memory. Shared Memory, Distributed Memory - PRAM model of parallel computation. Managing Peripheral devices - I/O Interface - Asynchronous Data Transfer - Modes of Transfer - Interrupt - DMA - I/O Processor - Data and control parallelism, Serial Communication.

Module III: Non-Linear Pipeline processors - Instruction pipeline design - Arithmetic pipeline Design - Pipeline control methods; Job sequencing, collision prevention and pipeline chaining; Super Scalar and Super pipeline Design. Instruction level parallelism, Advanced Architectures, multicore processing, multiprocessor systems.

References
- Carpinelli, Computer System Organization & Architecture
- Introduction to Parallel Computing 2/e Ananth Grama, George Karypis and Vipin Kumar, Anshul Gupta
- M. Sasikumar, et.al., “Introduction to Parallel Processing”, PHI, New Delhi, 2000

CS1612 Data Structures and Algorithms


References:
- Horowitz, Sahni, Rajasekaran; Computer algorithms - Galgotia, 1998
- Samanta D., Classic Data Structures, Prentice Hall of India.
- Levitin, Introduction to the Design and Analysis of Algorithms
- G.L. Heileman, Data Structures, Algorithms and Object Oriented Programming
- Horwitz, E and Sahni, Sartaj, Fundamentals of Data structures.- Galgotia
CS1613 Mathematical Foundations of Computer Science


Module II: Algebraic structures: Semigroups, Monoids, Groups, Subgroups, Symmetric groups, Groups homomorphism and isomorphism, Cosets and Lagrange’s Theorem, Normal subgroups, Permutation of groups and Burnside’s theorem. Finite –State Machines: Languages, representation of special grammars and languages, Finite state machines.

Module III: Graph Theory: Basic Concept of Graph Theory, Euler Paths and Circuits, Hamiltonian Paths and Circuits. Spanning tree, Probability: Axioms of probability, conditional probability, Baye’s theorem.

References:

- Bernard Kolman c, Busby & Sharon Ross, Discrete Mathematical Structures [PHI]
- C.J. Liu, Elements of Discrete Mathematics, MGH.

CS1614 Programming Paradigms


Module II: Object Oriented Programming in Java: Abstraction, Inheritance, Polymorphism, overriding, access specification, special features of Java, interfaces, packages, exception handling, finally clause, concurrent programming in Java: multithreading, event handling. Applets, threads in Applets, Java APIs, AWT: working with windows, Graphics, Text, using AWT controls, layout manager and menus. I/O streams, files, Introduction to RMI, Comparative study of C++ and JAVA.


References:

- R. Sebesta, Concepts of Programming Languages, Addison Wesley
Robert Lafore, Object-Oriented Programming with C++
Bruce Eckel, Thinking in Java
J. Reynolds, Theories of Programming Languages, Cambridge University Press.

CS1615 Computer Networks


References

Behrouz A Forouzan, Data Communications and Networking, McGraw-Hill, 2006
Brijendra Singh, Data Communication and Computer Networks, PHI, 2011.
Tanenbaum Andrew S., Computer Networks, TMH.
Comer, Computer Networks and Internet with Internet Applications, PHI, 2009.

CS1616 Data Structures & Algorithms Lab

It is recommended to implement the algorithms in C++, making use of the features of object oriented programming.

A. The list of experiments shall include the following topics:
   1. Programs to implement Linked list.
   2. Implement stack as linked list, queue as linked list.
   4. Search in a BST.
   5. Traversals on binary tree.
   6. Deletion from BST.
   7. Depth first search and breadth first search on a graph.
   9. Divide and conquer algorithms for binary search, maximum and minimum.
   10. Divide and conquer sorting: Quick sort and mergesort.
12. Implement single source shortest path algorithm
13. Implement matrix chain multiplication
14. Find longest subsequence.
15. 8-queens problem.

B. A case study on either one of the following:
   (a) Applications of any data structure.
   (b) Using nonlinear data structures in real problems.
   (c) A package for insertion deletion and traversal on various data structures
   (d) Any software/utility making use of one or more data structures or algorithms covered in the syllabus for CS1612.
   (e) Any simple game using data structures and one or more of the algorithms covered in the syllabus for CS1612.
   (f) Investigations on advanced data structures like splay trees, tries etc.
   (g) Studies/investigations on applications of multi-way search trees
   (h) Studies/investigations on algorithms with reduced complexity for various applications.
   (i) Any study/development of programs which applies or enhances the skills acquired through CS1612.
   (j) Comparative study of various algorithms based on complexity estimations.

CS1617 Java Programming Lab

A. The list of experiments shall include and cover the following topics:
   1. Java implementation of classes, objects and methods.
   2. Using input output streams.
   3. Method overloading and overriding.
   4. Inheritance.
   5. Interfaces
   6. packages
   7. Abstract classes.
   8. Vectors and wrapper classes.
   9. Multithreaded Programming and inter-process communication.
   10. Programs to differentiate Java from C++
   11. Exception handling
   12. Applet Programming
   13. AWT components.
   14. Managing Input / Output Files in JAVA.
   15. Connecting to database using JDBC
   16. Including graphics, video and sound in web pages, including Java applets
   17. Layers & Image Maps

A. A case study on either one of the following or any other which exhibits the capabilities and features of Java.
   (a) Applications using GUI.
   (b) Demonstrate network programming features in java.
   (c) Generate any class library
   (d) Any simple game using data JAVA.
SEMESTER II

CS1621 Modern Operating Systems


References:

- Silberschatz A and Galvin P, Operating system Concepts, 6/e Addison Wesley.
- NIIT, Operating System Linux, PHI.
- EVInEMETH, Snyder, KHein Trent, Linux Administration Handbook, Pearson.

CS1622 Advances in Database Management

Module I: Introduction- purpose of database systems, views of data– data abstraction, instances and schemas, data independence, data models Database languages- DDL, DML, transaction management, storage management, database administrator, database users, overall system structure. Relational data model- relational model concepts, keys, integrity constraints- domain constraints, key constraints, entity integrity constraints. Normalization theory- 1NF, 2NF, 3NF, BCNF, Multi valued dependencies and Fourth normal form– Join dependencies and Fifth normal form. Limitation of 4NF and BCNF.

Module II: ER data model – basic concepts, extended ER features, design of an ER database, reduction of an ER schema to tables. Relational algebra and calculus - tuple relational calculus, domain relational calculus. Object Oriented Database Management Systems (OODBMS) - concepts, limitation of relational model, need for OODBMS, composite objects, issues in OODBMSs, advantages and limitations of OODBMS, object model, object definition language, object query language.

Module III: Distributed database, distributed storage, types of distribution, heterogeneous and homogeneous DDBMS, functions of DDBMS, architecture of distributed databases, the design of distributed databases, distributed transactions, commit protocols – 2 phase and 3 phase protocols for distributed databases. Multimedia
database—growth of multimedia database, applications, contents of MMDB, designing. Spatial database and geographic database information systems, genome database—genomics, genome expression, proteomics, knowledge databases—deductive and semantic databases.

References


CS1623 Object Oriented Analysis and Design


References:

- Ambler, Scott W., The Object primer: the application developer’s guide to object orientation and UML. 2 rev ed. - Cambridge University Press.
- Booch, Grady; Rumbaugh, James; Jacobson, Ivar., The Unified Modeling Language user guide. - Pearson Education Asia, 2002.
CS1624 Graphics & Multimedia Systems

Module I: Introduction, output primitives and algorithms- DDA(Digital Differential Analyzer Algorithm) line drawing algorithm, Bresenham’s line drawing algorithm, midpoint circle drawing algorithm. Two Dimensional Geometric Transformations, Window to View port Transformations, Clipping, 3-D Display Techniques, 3-D transformations, Hidden Surface Removal Methods-Back face Detection, Depth Buffer Method, Scan-line Method.


References:

- Khalid Sayood, Introduction to Data Compression, Morgan Kaufmann Publishers
- Nelson, The Data Compression Book, MGH.
- Ralf Steinmetz KlaraNahrstedt, Multimedia Applications, Springer International Edition
- Judith Jeffcoate, Multimedia in Practice: Technology & Applications, PHI

CS1625 Optimization Techniques


References:

- Kanthi Swaroop, P.K Guptha, ManMohan, Operations Research
- Sharma J.K., Mathematical models in operations research, TMH.
CS 1626 Mini Project & Seminar

(a) Mini Project

Each student has to do a mini project in the college lab itself. Mini project will give an opportunity for
the students to prepare for main project and to achieve some of the objectives of the main project. This can
also be used as an opportunity for producing and distributing socially useful softwares.

The projects can be done individually or as a group of maximum three members. The type and scope
of the project shall be completely flexible. No restrictions shall be placed on the student on the choice of
platform/tools/languages to be utilized for their project work. The only constraint is that considerable effort
should be involved and there should be some technical work in the work. No value shall be placed on the use
of tools in the evaluation of the project.

There should be an internal supervisor for each student. An external supervisor can also be permitted if
the students opts for attachment to external organizations. But the external supervisor shall not be allowed to
assess or evaluate the work.

(b) Seminar

Each student has to present a seminar on current and emerging topics in and around computer science. The
students shall familiarize literature search, and practice assimilating of knowledge from published literature on
current and emerging topics and to communicate the same in their own words through a technical presentation.

A report of the seminar shall be prepared containing the details of literature search, scope, relevance, working
principles etc. shall be submitted by the student.

CS1627 Databases & Web Applications Lab

A. Databases:

1. SQL statements for creating, listing, dropping, checking, updating tables.
2. Record manipulation using insert, delete, update.
3. Experiments on the use of keys.
4. Queries with expressions.
5. Queries on aggregation, grouping.
6. Queries with substring comparison and ordering.
7. Queries to find values on different conditions and constraints.
8. Nested queries
9. Renaming attributes and joined tables
10. Experiments on other features covered in the course CS1622 Advanced Database Management

B. Web Programming lab experiments shall include the following topics.

1. Creating animated gifs, simple flash animations
2. Building Cascading Style Sheets
3. DHTML.
4. Creating and browsing XML database
5. Installing VRML plug-ins and viewing VRML source files
6. HTML forms and Fields
8. Installation and configuring of any Web Development platform like J2EE, WAMP/LAMP, .NET etc.
9. Development of web sites involving a variety of tools
10. Connecting to databases from web page.
11. Development of dynamic web pages with updation of database content.
12. Exercises for developing applications with database connectivity, making use of variety of tools and features covered in the syllabus of CS1622 Database Management and CS1615 Programming Paradigms.

C. Case study: Development of a web application with database and dynamic operations.

**SEMESTER III**

**CS1631 Data Mining & Warehousing**

**Module I:** Introduction to Data Mining. Different kinds of data and patterns that are mined. Technologies used. Applications, Major Issues. Data Objects and Attribute Types, Basic Statistical Description of Data – Data visualization, Measuring Data Similarity and Dissimilarity. Data Pre-processing, Data cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

**Module II:** Basic concepts of Data Warehousing. Data warehousing modeling: Data cube and OLAP – Data warehouse design and usage. Data Warehouse Design and usage. Data warehouse Implementation. Data cube Technology. Classification, Decision Tree Induction, Bayes classification, Rule based classification, classification by back propagation.


**References**

- Data Mining : Concepts and Techniques , 3rd ed., J Han, M Kember, J Pei, Morgan Kaufman
- Yazdani Sima, Wong, Shirley, Data Warehousing with Oracle, Addison Wesley.

**CS1632 Distributed Systems & Cloud Computing**

**Module I:** Characteristics of distributed System: Examples of distributed systems – issues in the design of distributed system. System models: Architectural models and fundamental models. Distributed objects and remote invocation: communication between distributed objects – remote procedure call – Events and notification. Operating system support: Operating system layer – protection – processes and threads-communication and invocation – Operating system architecture security: Overview of security techniques

**Module II:** Distributed file system: File service architecture - network file system- Andrew file system-recent advances Transactions and concurrency control: nested transactions-locks-optimistic concurrency control-comparison of methods for concurrency control-flat and nested distributed transactions- distributed deadlocks-transactions recovery. Replication System model and group communication- fault tolerant services-transactions with replicated data

**Module III:** Cloud Computing – Overview, Layers and types of cloud, Uses of Cloud; Components of Cloud Computing - Software as a Service, Platform as a Service, Infrastructure as a Service, Identity as a Service; Data storage in the cloud: Understanding, Advantages and Disadvantages of Cloud-Based Data Storage; Disaster Recovery – understanding threats; Service-Oriented Architecture – understanding SOA, Web services are not web pages
References:

- George Coulouris, Jean Dollimore and Tim Kindberg, Distributed Systems: Concepts and Design - Pearson Education
- Andrew S Tanenbaum and Marten Van Steen, Distributed Systems: Principles and paradigms – Pearson Education
- Venkatakrishna&etal, Principles of Grid Computing – Concepts And Applications, Ane Books

CS1633 Information Security


References

- Charlie Kaufman, Radia Perlman, Mike Speciner, Network Security- Private Communication in a Public World, Pearson Education
- Paar Christof, Pelzl Jan, Understanding Cryptography, Springer

CS1634 Compiler Design


Module II: Compilers- Compiler structure, compiler construction tools, Phases of compiler, Finite Automata, Push Down Automata (PDA), Non-determinism and NFA, DPDA and PDAs and languages accepted by these structures. Grammars, Languages-Types of grammars. The relationship between types of grammars, and finite machines. Push Down Automata (PDA) and Context free grammars (CFG). Lexical analysis: Specification and recognition of tokens, regular expressions and regular languages. LEX package on Unix. Conversion of NFA to DFA. Minimizing the number of states to DFA.

Text Books:


CS1635A Big Data Analytics


Module II: Big data technologies, distributed data processing, big data processing requirements, Hadoop, Components of Hadoop – The Hadoop Distributed File System, Hadoop MapReduce and Hadoop Common Components. Application Development in Hadoop – Pig, Hive. HBase. Getting Your Data into Hadoop – Basic Copy Data, Flume. NoSQL, CAP theorem.


References

- Krish Krishnan, Data Warehousing in the age of Big data, Morgan Kaufman.
- Understanding Big Data- Analytics for Enterprise Class Hadoop and Streaming Data. Chris Eaton, etc.
- Big Data Now- O’Reilly.
- David Loshin, Big data Analytics, Morgan Kaufman.

CS1635B Digital Image Processing

Module I: Image fundamentals, picture as collection of pixels, binary, grayscale, color images, color models, space requirements. Applications of image processing, Image sampling and quantization, neighbours of a pixel, adjacency, connectivity, boundaries, distance measures, image operations on pixel basis. Spatial domain operations: image negatives, log transformations, histogram Fundamental steps in digital image processing, components of an image processing system.

Module II: image enhancement using histogram equalization, histogram matching, image subtraction, averaging, spatial filters. Smoothing, sharpening filters Enhancement in frequency domain, two dimensional DFT, frequency domain filters. Basic concepts of discrete wavelet transform, advantages of frequency domain operations. Generating spatial mask from frequency domain.

References


CS1635C Machine Intelligence

**Module I:** Introduction to machine intelligence: Turing Test, Knowledge engineering, Knowledge acquisition, procedure for knowledge acquisition, knowledge base, inference engine, knowledge representation. Procedural vs Declarative Representation, Domain modeling, Different knowledge representation schemes- semantic net, frame, script.

**Module II:** AI and search process: need for heuristics, search methods: blind search and heuristic search, Depth-first, Breadth-first, Best First, Hill-climbing, Game Playing: Min-Max, alpha-beta pruning. Reasoning, expert systems, Rule Based Expert Systems, basic characteristics of expert systems, examples (MYCIN, DENDRAL-concepts only), applications of expert systems, robots, software agent

**Module III:** Uncertainty: Introduction, Non-monotonic and Monotonic Reasoning, Confidence Factor, Bayes Theorem, Non-classical logics, Default Logic, Bayesian Networks, Fuzzy Logic, language and language Processing: Speech Processing; speech coding, speech recognition, speech synthesis; Natural Language Processing: general concepts and issues, ambiguity in natural languages.

References

- Rich, Night, Nair, Artificial Intelligence, MGH.
- D.W. Patterson, Introduction to Artificial Intelligence and Expert Systems, PHI.

CS1635D Programming for Portable Devices

**Module I:** Wireless Application Protocol, WAP architecture, WAP client, WAP browser, advantages, applications, WML, language constructs, variables and contexts, tasks and event, WML user interaction, templates, text and image formatting, handling audio, wireless application services, IrDA protocols, WAP gateway, extension for mobile devices, synchronization of mobile devices and wired network.


**Module III:** Case study of operating system for mobile devices: architecture, memory management, interrupt handling and multi-tasking in Blackberry OS, Coding with Blackberry JDE, Blackberry App, user interface, iOS6 architecture, frameworks, SDK, MVC architecture, overview of objective C.

References

- C.S.R. Prabhu, Mobile Computing, Universities Press
- Martin Frost, Learning WML and WML Script, O’Reilly.
- REto Meier, Professional Android Applications Development, Wiley.
- Donn Felker, Android Applications Development, Wiley International
CS1636 Network Administration Lab

- Installing and configuring Linux, disk partitions, installing and removing packages.
- Linux Administration, identifying administrative files configuration and log files.
- Managing user accounts, changing permissions and ownerships, creating and managing groups, disabling user accounts.
- Boot loader management, lilo, grub, configuration - ckconfig, ntsys.
- Checking and monitoring system performance.
- File security, permissions.
- Getting system information.
- Study of different types of cables – coaxial, UTP, etc.
- Study of Hub, Switch, Repeater, Gateway.
- Installing a network card.
- Connect a computer to a LAN and use the Internet/copy file.
- Basic network setup, hostname, ip address etc.
- Setting up an internet connection, setting up wireless connections.
- Communication – mesg, talk, write, wall, finger, ping, tracerout.
- Configuring Static IP in Windows and Linux.
- Study of Network commands in Linux - ping, traceroute, nslookup, etc.
- Study of Network files used in Linux.
- Experiments using WireShark packet sniffer.
- Study of Ethereal as an analysis tool.
- Use of Ethereal to analyze HTTP, SMTP, FTP.
- Configure a network using Packet Tracer Software.
- Configure a network with any routing protocol using Packet Tracer Software.
- Study of different Proxy, Web Server softwares.

CS1637 Distributed Computing Lab

A. Experiments shall include and cover following topics.
- Implementing Java RMI, RPC
- TCP, FTP, UDP programming
- Exercise on Servlets
- Exercise with EJB
- Client-server programs
- Experiments on cloud storage
- Experiments on network traffic management in simulated distributed environments
- Installation and setting up of Hadoop.
- Exercises on development using Hadoop
- Exercises on practicing programming for portable devices on platforms like Android

B. Case study on developing a cloud/distributed environment/developing distributed applications/developing applications for mobile devices/study on using various features of Hadoop/CUDA etc.
SEMESTER IV

CS1641 Research & Technical Writing

Module I: Research: Objectives and types of research: Motivation and objectives, Defining and formulating the research problem, Importance of literature review, Identifying gap areas from literature review Research-Definitions & types of research. The Scientific Method- Observation- Questions- Hypothesis- Experimentation-Critical Communication-Presenting and publishing research work- seminar, workshop, symposium, conference.


Module III: Introduction to Python: Environment Variables, Collections- Lists, Tuples, Sets, Dictionaries, Sorting Dictionaries, Copying Collections, String Operations, The format Method, Functions- Defining Your Own Functions, Optional Parameters, Passing Collections to a Function, Variable Number of Arguments, Scope, Mapping Functions in a Dictionary, Lambda, Closures, Iterators, Text, Binary Handling: iteration protocol, iterable objects, generators and generator expressions, data processing pipelines. Creating and using classes in Python, Custom Exception Classes, pydoc, Exception handling Multiple Exceptions, Python & CGI; Python Interacting with databases

References

- Kottwiz, LaTEX for Beginners.
- Kopka, A guide to Latex.
- Kothari, C.R., Research Methodology: Methods and Techniques. New Age International Publishers

CS1642A. Parallel Computing Algorithms

Module I: Parallelism, Temporal Parallelism, Data parallelism, inter-task dispatching, Instruction level parallelism, delay in pipelining, superscalar processors, VLIW processors. Multithreaded processors, structure of parallel computers, Classification of parallel computers.

Module II: Vector computers, IRAM, array processors, shared memory, distributed shared memory. Interconnection networks, message passing parallel computers, cluster of workstations. Use of MPI in clustering, dedicated high performance cluster, on-demand, high-throughput, data intensive computing.


Reference:

- Parallel Computing Algorithms,
- Sasi Kumar, Shikhare & Ravi Prakash, Introduction to Parallel Processing, PHI, 2006.
- Quinn, Parallel Computing: Theory and Practice, MGH.
CS1642B Bioinformatics


Module II: Analyzing DNA sequence, IUPAC code for DNA sequence, palindromes in DNA sequence, RNA sequence analysis; FASTA format. Sequence analysis/Alignment: DNA sequence, RNA sequence, Protein sequence, sequence alignment classifications, analyzing protein sequence, human insulin sequence, Scoring Matrices – PAM, BLOSUM; Sequence Alignment: Introduction to Sequence Comparison - Pairwise Alignment Method (DOT PLOT method) and Multiple Analyses of Protein Structures; STS and EST sequences; DNA Microarray.

Module III Standard genetic code Biological databases: different types, typical data banks, GenBank, Swissprot, PDB; molecular visualization tools: Rasmol, Swiss PDB viewer; Searching PubMed, Protein information site: Expasy; retrieving protein/DNA sequences, Exploring the Human Genome. Sequence Analysis Tools – BLAST, FASTA; Prediction Tools – GENSCAN, SNP; Importance of Perl language in Bioinformatics; system biology Applications of Bioinformatics in Biodiversity, Human Genetics, Gene Therapy, Agriculture (overview only)

References

- Dr. K Mani & N Vijayaraj, Bioinformatics: A practical approach, Aparna Publications
- Dan E Crane and Michael L Raymer, Fundamental Concepts of Bioinformatics, Pearson Education
- Selzer-Marhofer-Rohwer, Applied Bioinformatics – an introduction, Springer
- ZhumurGhosh, BibekanandMallick, Bioinformatics – Principles and Applications, Oxford Higher Education

CS1642C Neural Networks & Fuzzy Systems


Module III: Fuzzy systems: uncertainty and information, fuzzy sets and membership, fuzziness, classical sets and fuzzy-sets-operations, properties, properties of membership functions, fuzzification, defuzzification to crisp sets, fuzzy logic, applications. Fundamentals of genetic algorithms, creation of offsprings, reproduction, genetic modeling, operators. applications of genetic algorithms.

Text Books:

- B. Yegnanarayana, Artificial Neural Networks - PHI
- Philip D. Wasserman, “Neural Computing – Theory and Practice”, Van Nostrand and Reinhold
- Sivanandan, Deepa, Principles of soft computing, Wiley India.
- Sathish Kumar, Neural Networks: A classroom approach, MGH.
- Ross, Fuzzy logic with engineering applications, Wiley India
CS1642D Embedded Systems


References
- Raj Kamal Embedded Systems- Architecture, Programming and Design, 2nd edition, TMH.
- Rao B. Kanta, Embedded Systems, PHI.
- Elecia White, Making Embedded Systems, O’Reilly

CS1643 Major Project

Major project work shall be done individually by each student under the guidance of a faculty member from the department. The Project proposals and synopsis submission shall be done in third semester itself. It is advisable to select the project topic and area keeping the following objectives in mind:

1. The project work shall give enough opportunity for the students to apply some of the skills and knowledge earned through the theory courses.

2. The student shall get an exposure in developing industry type applications/utility software for computer systems or mobile devices/in studying and analyzing theoretical concepts and presenting comparative analysis of state-of-the art techniques/in developing new or improved algorithms/in the use of soft computing techniques in selected area/discipline.

The students need to do the following activities:

1. The candidate shall submit proposal for different projects before the evaluation team. The team shall select and finalize one of the proposals. However if all proposals are not acceptable, he may be asked to submit new/modified proposals. The candidate shall prepare and submit a synopsis of the accepted proposal. A record of accepted synopsis of each candidate shall be maintained in the department.

2. A detailed study of the requirements and feasibility of the proposed work shall be conducted by the candidate with the help of the project guide. A study phase report shall be presented before the evaluation team within one month from the beginning of project work. The design of proposed work shall be completed and presented before the evaluation team. The design shall be finalized with suggested corrections/updations.

3. The developed software/algorithm shall be implemented and demonstrated before the internal evaluation team. A short presentation explaining the proposed work and experimental results shall also be made. The project Report shall be finalized only after the internal presentation- after correcting/updating the document based on the comments from internal evaluation team.

***
UNIVERSITY OF KERALA

SCHEME AND SYLLABUS OF
M Sc COMPUTER SCIENCE
(2016 ADMISSION ONWARDS)